40 Box (C)

POPULAR Computing WEEKLY

35p 6 January 1983 Vol 2 No 1

This Week

Spectrum software
John Scriven takes another look at some of the latest games for the ZX Spectrum. See page 12

Data transfer
Kevin Griffiths presents a routine to transfer data from one program to another on the 16K ZX81 on page 23.

Software library
David Kelly talks to Alec Fry founder of the Sinclair Owners' Software Library See page 11

Dragon graph
G Morton explains how to represent data on an x,y scale using a simple graph plotting routine on page 25

★STAR
Missile Command on Spectrum by Chris
GAME★

News Desk

Timex-Sinclair 1000 – already selling well in the US

Spectrum to go on sale in US?

A US version of the Sinclair ZX Spectrum is due to go on sale in America in the first quarter of 1983, possibly as early as January

The machine will be marketed and sold exclusively by Timex in the US

The American company's first product — the TS1000 (a 2K version of the ZX81) — has been a runaway success since it was first launched in August

Timex has now exceeded the necessary sales threshold which it has to reach to obtain an exclusive licence to sell computer products based on Sinclair technology in North America Under the agreement between Timex and Sinclair Research, Sinclair is now required to

Continued on page 6

Bug-Byte goes retail

BUG BYTE is set to become the first major software house to cease trading by mail-order

As of March the company will only be selling its range of software cassettes through retail computer shops and chain stores

Bug-Byte's decision to phase out mail-order selling emphasises the extent to which the micro-computer industry is now looking to the High Street for most of its trade

"At the same time as sales to the retail trade are increasing we are seeing a dwindling mail-order demand" explained Bug-Byte co-founder Tony Milner

"Dealing with postal sales uses up 60 percent of our workforce but only brings in about 20 percent of the turnover

"Our last mail-order advertisement is due to appear in March and from then on we shall phase out postal selling We are hoping that this will help our dealers — knowing that they will become our sole outlet"

Classified

Computer Swap 01-930 3266

Free readers entries to buy or sell a computer
Ring 01 930 3266 and give us the details

ZX81 + 16K RAM £100 of software magazines and manuals £75 ono Tel Wiveliscombe 23783

DRAGON 32. Brand new £199.50 Mr Delroy Tel Leeds (0532) 056879 evenings.

ATOM 8K plus 88 new keyboard VIA joystick interface 2MHZ mains power supply £130 Mr Clarke Leeds 732580

ATARI 800 £200 approx or swop for my Super Board III system. Tel 01-200 7028 after 6 pm

ATARI 800 Offers to Tom Klimes Tel Bourne End 26576 after office hours or weekends

ATARI VCS. Excellent condition hardly used complete with Combat and Asteroids cartridges £120 ono Tel Whaley Bridge 3206 (evenings and weekends)

TI 99/4 with extended Basic Video Chess and video games cartridges cassette lead 5 issues of 99ER magazine and over 20 programs, £220 Tel North Weald 2115

DRAGON 32 plus cassette lead, £165 Barnsley 42541 evenings after 5 pm

SPECTRUM OWNERS

All the software you'll ever need. The best Spectrum programmes for a once-only payment of £10 plus a hire fee of £1.25 per tape.

JOIN TODAY Send £10 for life membership and 5 of free tapes to

SPECTRAL SOFTWARE LIBRARY
33 Charlecote Road Poynton
Stockport Cheshire Sk12 1DJ
or send SAE for details

COMMODORE PET 2001 series 32K cassette recorder with sound, program toolkit 130 software cassettes manuals and books £325 ono K Berwick Erith (30) 36358

VIC20 with data cassette super expander Galaxians joystick and lots of software £190 Tel Coalis 38811

Got a **DRAGON 32** or **TANDY TRS80** colour computer? Then you need your own monthly magazine **RAINBOW** for colour computer users. Send £1 95 and large 28p s.a.e. for sample issue to **ELKAN ELECTRONICS (Dept PWK FREEPOST 28 Bury New Road Prestwich Manchester M25 0LZ Telephone 061 798 7613 (24-hour service)**

SPECTRUM ASSEMBLER
Enter the world of the Z-80! Full 2 pass assembler with labels — all opcodes — 11 powerful directives — easy program editing — 16/48K — manual. Written and tested by professionals. **£8.95 Cheque/PO to C Newport, 67 Camlet Way Hadley Wood Herts**

16K ZX81 With software in good condition £90 Mr A Hill 18 Toft Avenue Grays, Essex

Continued on page 28

BATTLESTAR IS COMING — SOON

POPULAR Computing WEEKLY

6 January 1983 Vol 2 No 1

The Team

Editor
Brendon Gore

News Editor
David Kelly [01-930 3271]

Sub-editor
Ninette Sharp

Editorial Secretary
Theresa Lacy

Advertisement Manager
David Lake [01-839 2846]

Advertisement Executive
Alastair Macintosh [01-930 3260]

Managing Editor
Duncan Scot

Publishing Director
Jenny Ireland

Popular Computing Weekly,
Hobhouse Court, 19 Whitcomb Street,
London WC2
Telephone 01-839 6835

Published by Sunshine Publications Ltd

Typesetting, origination and printing by
Chesham Press, Chesham, Bucks

Distributed by S M Distribution
London SW9 01-274 8611 Telex 261643

Subscriptions
You can have *Popular Computing Weekly* sent to your home
UK Addresses

26 issues	£9 98
52 issues	£19 96

Overseas Addresses

26 issues	£18 70
52 issues	£37 40

How to submit articles
Articles which are submitted for publication should not be more than 3 000 words long The articles and any accompanying programs, should be original It is breaking the law of copyright to copy programs out of other magazines and submit them here — so please do not be tempted

All submissions should be typed and a double space should be left between each line Please leave wibe margins

Programs should whenever possible be computer printed

We cannot guarantee to return every submitted article or program so please keep a copy If you want to have your program returned you miss include a stamped addressed envelope

Accuracy
Popular Computing Weekly cannot accept any responsibility for any errors in programs we publish although we will always try our best to make sure programs work

This Week

Editorial

Anyone who has ever looked inside a Sinclair printer will know that it is more complex than it appears from the outside Anyone who has ever taken a Sinclair printer apart will testify to the difficulty of putting it back together.

The Sinclair printer is a mass of little white plastic wheels and cogs, bestrewn with wires and connectors The electric stylus, which burns through Sinclair's aluminised paper to form letters and characters, is attached to a whirling rubber band.

But for all the intricacy of the Sinclair printer's design the end result is at best barely adequate Burnt carbon from the aluminised paper tends to clog up the works, causing already faint listings to become completely illegible

Mind you, even at £59 95 the Sinclair printer is still considerably cheaper than its rivals, so it is perhaps a little unfair to expect pristine copy every time

But everyone who has suffered from the vagaries of the Sinclair printer will be glad to know that Sinclair is rumoured to be working on a four colour printer that will sell for around £70 I should emphasise that this is only a rumour, though Sinclair is known to be developing a printer of some sort I shall await its appearance with anticipation

Next Thursday

At last the mystery can be revealed Battlestar, a unique computer-moderated, play-by-mail game, starts next week. To enter Battlestar, a game which is exclusive to readers of *Popular Computing Weekly*, simply buy next week a copy.

US Spectrum

Continued from page 1

wind up its US computer selling operation.

The American division of Sinclair placed its last computer advertisement in September and was then given 90 days to conclude all outstanding business. But Sinclair's US office will be retained to market the company's flat-screen tv when it becomes available.

A spokesman for Sinclair Research commented: "The Timex licence is now fully exclusive in the North American market and sales of Sinclair's own-brand computers there are now prohibited.

"The decision to sell a version of the Spectrum over there is ultimately Timex's but it must happen in the first quarter of 1983, possibly early in January."

Midwich joystick

Joysticks from Midwich

MIDWICH Computers has introduced a range of joystick units to its add-on collection.

For use with the Dragon 32, Acorn, BBC, ZX81 and Spectrum machines, the analogue joystick potentiometers have a life expectancy in excess of 200,000 operations.

Since neither of the Sinclair machines are provided with a built-in analogue/digital converter, Midwich has also produced a high-speed joystick controller board.

The units are available from Midwich Computers, Rickinghall House, Hinderclay Road, Rickinghall, Suffolk and are priced as follows (including VAT): Dragon 32 £15.98 per pair, Acorn BBC £13.00 per pair, ZX81/Spectrum £15.98 per pair.

Dragon lament

A SMALL bug crept into the Dragon *Singalong* program in our December 16/23 issue. Line 40 should have read:

```
40 CLS X$="ABCDEFG"
```

Imagine software

SENIOR staff at the Liverpool based software company Bug-Byte have broken away to set up their own firm.

Dave Lawson, former software manager at Bug-Byte and Mark Butler, until recently sales manager at Bug-Byte, have formed a new company — Imagine Software. Also involved in the new venture is Bug-Byte's former head programmer, Eugene Evans.

The first fruits of Imagine Software have appeared in the form of *Arcadia*, a new game for the Spectrum and Vic20. Two more games will follow on January 14.

"What we are doing now is entirely different from Bug-Byte," said Dave Lawson. "We hope to be able to produce at least two new games each month — and all our software will be original rather than versions of existing arcade games."

At the moment Imagine software is available only by mail order. By the end of January, however, the program will be available in the high street chain stores and specialised computer shops.

Bug Byte remains undaunted by the departures.

"I gather some of our old people have set up an outfit just up the road," said Bug-Byte's Tony Milner. "We are not at all worried — if anything we have become more efficient since they left.

"We're still good friends. They are not any competition yet but it will keep us on our toes," he said.

Young Computer Brain 1982

Derek Reynolds (left) and Peter Hall

A FOURTEEN year old schoolboy from Newcastle-upon-Tyne has been chosen as Young Computer Brain of 1982.

Derek Reynolds' winning program — designed to help handicapped people to teach themselves to use a computer — was selected from over 320 entries. As the winner he receives £2,000-worth of computer equipment from Commodore Business Machines and a trophy from the *Sunday Times Magazine*, joint sponsors of the event. The trophy was presented by Peter Hall, Chairman of the Council of the British Computer Society at a ceremony held on December 13.

The competition was divided into three classes. Derek Reynolds was also chosen as winner in the 13-14 age section.

Rachael Gooberman from Oldham won first prize in the under 13s category for her entry on how computer-aided design could be applied to police Identikit methods. Lionel Tun from Mitcham won the 16-18 section with a program to provide computer-used sleep therapy.

The aim of the competition, held every year, is to encourage young people to use computers to benefit society.

Scottish show

THE *Personal Computer World Show* is travelling north.

The *Scottish Personal Computer World Show* is to be held on April 16-18 (Saturday to Monday) at the MacRobert Pavilion, Ingliston, Edinburgh. More details from Jenny King on 01-486 1951.

High Street training is 'essential'

DEREK Moon, managing director of Currys Micro Systems has hit out against selling microcomputers without specialist sales staff and aftersales support.

"The market is not ready for cash and carry computers," he said. "Uncontrolled selling of home computers will cause retailers problems they haven't begun to imagine. If the shop staff are ill-informed or misinformed there will be a queue of customers dissatisfied with the retailer and disenchanted with the idea of home computing."

In line with this thinking Currys will only at present be selling microcomputers in high street branches in proximity to their nine Micro-C specialist computer shops. This will ensure that customers will not have to go far to sort out any problems they may have. By the New Year 37 of Curry's 512 branches will be selling micros.

● Dixons is to send over 300 of its staff on a two-day intensive microcomputer training course. The 20-hour scheme will teach computer selling and also simple program writing. Dixons already sell the Commodore Vic20 computer and will shortly begin sale of the Camputers Lynx.

Dragon schools' software

DRAGON Data plans a move into the educational software market early in the new year.

Initially the company is to produce a range of programs aimed at 4- to 11-year-olds. The software will be split into two groups devoted to teaching numeracy and literacy.

Later the catalogue will be extended with material for the 12- to 15-year-old range.

This expansion into educational software is to compliment the company's plans to produce a schools version of its Dragon32 microcomputer. The model, which will have a built-in RGB monitor and cassette player, is currently under development.

Joining the majority

Re The Monkey Puzzle' November 18 page 35 If the question was formulated by Seymour Papert exactly as quoted by your contributor, then it is not surprising that three-quarters of the students asked by Papert gave 'wrong' answers As the question stands, the answer given as correct (that the rock goes up) is actually incorrect

You stated that the monkey and the rock are of equal weight In this case in order to balance one another as also stated, *both* must be resting partly on the ground, or *both* must be completely clear of the ground The question asks whether the rock moves up, or down, or stays still thus implying that it is free to move down, which means that it cannot be resting even partly, on the ground. This means, in turn, that the monkey also must be completely clear of the ground, with the *whole* of its weight already on the rope.

Starting to climb up the rope will have no effect on the weight of either monkey or rock so the rock will stay where it is

S Kane
66 Haw Road
Co Antrim

Boris Allan replies: by pulling on the rock (to raise himself) the monkey effectively applies a turning force to the pully (ie a "couple") and thus the rock rises. The solution is a case of action/reaction, and the monkey rises at exactly the same rate as the rock.

On a winning ticket

May I express through your columns, my thanks to David Lawrence for his *Working Spectrum* a copy of which arrived this morning (November 17) As I ordered it on November 12 this must set something of a record in the world of micro-computers

On a first swift look through the book it would appear to be invaluable and credit must also go to the designer for the very clear way in which the information is presented No large chunks of indigestible text

Thank you and the team for *Popular Computing Weekly* I'm a committed fan and look forward to each issue

Marion Taylor
504 Ben Jonson House
Barbican
London EC2Y 8DL

Niggardly bug examples

Dare I say that the examples of Spectrum bugs offered by your correspondents (so far) have been niggardly almost insignificant examples.

This one produces an entire incomprehensible screen display First enter

```
10 PRINT "xxx" GOTO 10 and
   RUN it
```

The screen will fill up and the computer stops to ask *Scroll?* Press both shift keys together and then *Enter*

Can anyone tell me what's going on?

John Bloxham
18 Lea Close
Stratford-upon Avon
Warwickshire CV37 9JS

When a bug is not a bug

David Edwards's Spectrum "bug" reported in your December 9 issue is not only not a bug it is actually documented on page 114 of the Spectrum manual, which fully explains the phenomenon

For the uninitiated, 6 in extended mode generates a "paper yellow" control code sequence, ie *Chr$17* + *Chr$6* Pressing *Delete* once deletes the *Chr$17* leaving *Chr$6*, which reference to the character code chart on page 183 will show is the control character corresponding to a comma in a *Print* statement hence the cursor moves to column 16

None of the other colour codes (0-5 and 7) have any meaning to the tv display, hence they are displayed as a question mark

All this does is illustrate the interesting fact that *Delete* works on control code sequences starting with the first code and working through to the last rather than the other way round as with normal displayed characters Incidentally 9 in extended mode sets the *Bright* attribute, not colour white as stated by Mr Edwards

There seem to be very few "real" bugs in the Spectrum, most of those reported are interesting quirks with little or no practical significance My contribution to the "real but avoidable" category is that *Clear* does not do a *Restore*, contrary to the manual This problem is overcome by the good practice of putting a *Restore* before any critical *Read* statements

Kevin Gordon
41 Fennel Crescent
Broadfield
Crawley
West Sussex

Bugged up and interesting

I think I have found another bug in the Spectrum — an interesting one Normally when the computer gives an error code, the cursor disappears Then, when a key is pressed, the message disappears and the cursor returns But the following program gives a different result

```
10 INPUT 3 a$
```

When the program is run, the error message "J Invalid I/O device, 10 1" is given — but the cursor appears at the end! Any typing done then will appear on screen at the same time as the error report, which cannot be deleted This does not disappear until *Enter* is pressed when the message is removed before syntax checking starts

Has anyone else noticed this fault? It seems to arise from the fact that you are telling it to accept data from the printer rather than the keyboard

Bill Longley
388 Ipswich Road
Colchester
Essex CO4 4EX

In a minority with only 16K

As a reader of your magazine since No 1, I wonder if you (or anyone else) can explain to me why the authors of programs, and especially software companies who retail the various program cassettes, always assume that the maximum capacity of the ZX81 is only 16K

I have a ZX81 32K There must be many thousands like me, and also many thousands with 64K Ram packs, yet there do not seem to be any cassettes on the market to take advantage of this

The real reason I am writing is that recently I purchased a ZX81 machine code compiler only to find out that it just had variables A-Z, no strings and on arrays unless you used *Peek* and *Poke*

As I wanted it to process the loops in programs using strings and arrays, but am not too familiar with the *Poke* command, the compiler couldn't help me a lot The reason given in the instruction leaflet was lack of space as the ZX81 only had 16K Ram

I would like to gamble that if a check could be made on all ZX81 users throughout the country that the 16K Rammers would be in the minority

J Ashbourne
212 Cherry Sutton
Hough Green
Widnes

A moot point. It is a gamble I would like to take, but the thought of conducting a nationwide survey of ZX81 owners is a little daunting

If you feel 16K owners are a majority/minority, please let us know.

Request for Vic adventures

After seeing the letter in your September 23 issue about Vic adventure games, my friends and I decided to write to you asking for some

We all own Vics and enjoy buying your magazine a lot

A M Smith
E Midwinter
P Eastman
N Oakly
21 Willingdon Park Drive
Eastbourne
East Sussex

There was not a massive response to our request for Vic adventures, but there was enough interest to justify futher action. We hope to run an adventure feature some time in the New Year.

If you have an opinion you want to express, or have spotted an error that needs correcting, write to Letters, Popular Computing Weekly, *Hobhouse Court, 19 Whitcomb Street, London WC2*

Missile command

A new game for the 16K Spectrum by Chris Wood

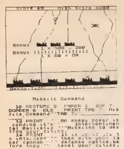
After a visit down my local arcade for ideas for games for my ZX Spectrum, I decided that *Missile Command* would be fast enough in basic. Below is an outline of the program

Lines

- **10 to 23** Set colours, and print instructions. Line 20 makes the cures into the word *Continue* to look neater
- **24 to 38** Creates the user definable graphics. Run when you get this far so that you know which ones to put in lines later
- **48 to 56** Defines the remaining variables and sets up the screen. Line 47 enables you to print on line 22.
- **90 to 140** The main game routine. Line 138 sends the program to the subroutine at 200 to check if a missile has been shot down if the 1(one) key has been pressed and there is still some ammunition
- **130** Scans the *Attributes* of the cities on line 20 to see if they have been hit by missile
- **179** Sends the program to the enemy satellite routine at 400
- **171 to 194** Works out the bonus for the remaining ammunition and cities according to the wave attack number
- **200** Checks to see if a missile has been correctly hit. If it has it decreases the number of missiles by one and to ensure the correct missile is stopped the position of the last missile replaces the destroyed one. A mushroom cloud is printed at the end of the missile trace
- **400 to 460** This is in effect a game on its own. It has had to be written like this to make it very fast and difficult and to avoid clutter by putting it earlier. It is last because the bonus is very large and there would be no point in making it too easy

Full instructions for playing are included in the program





Missile Command

```
10 RESTORE 9: PAPER 1: INK 7: BORDER 0: CLS : PRINT TAB 7;"Missile Command"''TAB 7;"_______________"
12 PRINT '"An enemy power is sending it's"'" Intercontinental Ballistic "'"Missiles to destroy your cities."
14 PRINT '"It is up to you to intercept "'" them with your laser cannon "'" defense system,before they "'" level your cities."
16 PRINT '"The controls are as follows"'"Up Down Left Right Fire"'" 7   6   5   8   1"
18 PRINT '"You get a bonus for the cities"'" you save and your unused"'" ammunition."
19 FOR B=1 TO 40: BEEP .1,RND*20: NEXT B
20 POKE 23617,210: INPUT FLASH 1;"PRESS ENTER TO ", LINE a$: POKE 23617,0
22 CLS : PRINT '"You cannot go below the line"'" and if you try you cannot"'" raise your sights again."'" So be careful and"
23 PRINT '"At the end of each wave an"'" enemy satelite will go across"'" the screen. if you can aim at"'" it your automatic jamming "'" transmitter will destroy it."''TAB 9;"GOOD LUCK"
24 PAUSE 500: LET HS=5000
25 POKE 23658,8: LET s3=0: LET s2=0: LET sc=0: LET bn=1
30 FOR q=144 TO 150: FOR n=0 TO 7: READ a: POKE USR CHR$ q+n,a: NEXT n: NEXT q
35 DATA 0,16,16,16,16,16,40,0,240,243,247,255,255,255,255,255
36 DATA 3,7,239,231,255,255,255,255,1,156,220,156,118,252,254,255,0,60,66,129,129,126,24,24
38 DATA 3,227,244,216,56,180,287,3,192,199,47,27,28,45,199,192
40 LET s=0
43 PAPER 5: INK 2: BORDER 6: CLS : PRINT AT 21,0; PAPER 4;TAB 31;" "
44 PRINT AT 0,0; PAPER 0;"     Score     High Score"
45 PRINT AT 0,9; PAPER 0; INK 7;sc;AT 0,26;hs
46 LET bn=bn+.5: LET r=3*(6-s): LET e=6
47 POKE 23659,1: POKE 23689,2: PRINT AT 22,0; INK 7; PAPER 0;TAB 31;" ";AT 22,0;"Ammunition ": FOR w=1 TO r: PRINT PAPER 1;"1";: NEXT w: POKE 23659,2
48 PLOT 0,30: DRAW 255,0
50 LET x=10: LET y=10: LET x1=x: LET y1=y
55 PRINT AT x,y; OVER 1; INK 0;"X"
60 DIM b(6): FOR i=1 TO 6: LET b(i)=-1+INT (RND*3): IF NOT b(i) AND i>3 THEN LET b(i)=-1
65 IF NOT b(i) AND i<3 THEN LET b(i)=1
70 NEXT i
71 PRINT AT 20,0;"  ";
75 FOR n=1 TO 6-s: PRINT INK 0;" BCD ";: NEXT n
80 DIM a(6): DIM d(6): FOR i=1 TO 6: LET d(i)=5: LET a(i)=167: NEXT i
85 DIM f(6): FOR i=1 TO 6: LET f(i)=INT (RND*40)*(0-s/2)+12: NEXT i
90 FOR j=1 TO 20
95 FOR i=1 TO e: PLOT f(i),a(i): DRAW b(i)*(INT (RND*5)),-d(i): LET f(i)=PEEK 23677: LET a(i)=PEEK 23678
100 PRINT AT x1,y1; OVER 1; INK 0;"X"
105 IF PEEK 23677<9 AND b(i)<0 THEN LET b(i)=-b(i): GO TO 115
106 IF PEEK 23677>245 AND b(i)>0 THEN LET b(i)=-b(i): GO TO 115
110 IF INT (RND*4)=4 THEN LET b(i)=-b(i)
115 IF x<17 THEN LET x=x+(INKEY$="6")-(INKEY$="7")
120 LET y=y+(INKEY$="8")-(INKEY$="5")
125 PRINT AT x,y; OVER 1; INK 0;"X": LET x1=x: LET y1=y
130 IF INKEY$="1" AND r>0 THEN GO SUB 200
135 BEEP .001,45-j/2
140 NEXT i: NEXT j
160 FOR n=2 TO 27-(5*s) STEP 5: FOR g=0 TO 2: IF ATTR (20,n+g)=42 THEN LET s=s+1: PRINT AT 19,n+g; INK 7; FLASH 1;"E": NEXT n: IF n>27-(5*(s-1)) THEN GO TO 170
162 NEXT g: NEXT n
170 GO SUB 400
171 PRINT AT 10,2;"Bonus ";: FOR v=1 TO 6-s: PRINT ; INK 0;"BCD";: BEEP .25,6-v: LET s2=s2+100*INT (bn): NEXT v
172 PRINT TAB 10;INT (bn);" x ";v-1;"00 = ";s2
175 PRINT AT 12,2;"Bonus ";: FOR v=1 TO r: PRINT ; INK 0;"1";: BEEP .1,-10: LET s3=s3+5*INT (bn): NEXT v
176 PRINT TAB 10;INT (bn);" x ";r*5;" = ";s3
177 LET sc=sc+s2+s3: LET s2=0: LET s3=0
178 IF sc>=hs THEN LET hs=sc
180 IF s=6 THEN GO TO 190
185 FOR U=1 TO 200: NEXT U: GO TO 43
190 PRINT PAPER 0; INK 7;AT 0,9;sc;AT 0,26;hs: FOR g=1 TO 48: BEEP .01,g: BEEP .01,48-g: NEXT g: INPUT "Another go? (Y/N)";a$
192 IF a$="Y" THEN CLS : RESTORE 9: GO TO 25
194 STOP
200 LET r=r-1: POKE 23659,1: POKE 23659,2: PRINT AT 22,12+r; PAPER 1;" ": POKE 23659,2: BEEP .01,40: FOR d=1 TO 6: IF INT (f(d)/8)=y AND 21-(INT (a(d)/8))=x THEN LET f(d)=f(e): BEEP .03,-10: PRINT AT x-1,y; INK 7; FLASH 1;" ": LET e=e-1: LET sc=sc+INT (bn)*20: PRINT AT 0,9; PAPER 0; INK 7;sc
205 NEXT d
210 RETURN
400 PRINT AT x,y; OVER 0;" ": LET x2=INT (RND*10)+3: LET x=10: LET y=10: LET x1=x: LET y1=y
405 FOR k=1 TO 29: PRINT AT x2,k;" GF"
406 PRINT AT x1,y1; OVER 1; INK 7;"X"
410 IF y<28 THEN LET y=y+(INKEY$="8")-(INKEY$="5")
420 IF x<17 THEN LET x=x+(INKEY$="6")-(INKEY$="7")
430 PRINT AT x1,y1; OVER 1; INK 7;"X": LET x1=x: LET y1=y
440 IF x1=x2 AND y1=k+2 THEN GO TO 460
445 IF x1=x2 AND y1=k+1 THEN GO TO 460
450 NEXT k: PRINT AT x2,29;"   "
455 RETURN
460 PRINT AT x2,k+1; FLASH 1;"EE"
470 LET sc=sc+500*INT (bn)
475 PAUSE 50
480 RETURN
1000 FOR n=0 TO 7: PRINT PEEK (USR "F"+n): NEXT n
```



Alec Fry at work with some of his staff

At the library in Liss

***David Kelly** talks to Alec Fry, founder of the Sinclair Owners' Software Library*

Mention the idea of a software library to some software producers and they turn puce and have to be restrained. They see the library as an institution set up specifically to deprive them of sales.

Other software companies are content to coexist with libraries and take an altogether different view.

Alec Fry runs one such venture — the Sinclair Owners' Software Library, based in deepest Hampshire. After only six months — the service started in July 1982 — the library has over 1,000 members and has more than 150 ZX81 and ZX Spectrum titles for hire.

"Last Easter I bought a ZX81," says Alec, "and I quickly realised that it was easy to spend as much on software as on the machine.

"So it seemed like a good idea to build up a stock of programs and start a library — I was surprised that nobody had done it before."

Membership of the Sinclair Owner's Software Library costs £8.50 or £9.50 a year, depending on whether you wish to hire ZX81 or Spectrum cassettes. For this you get a quarterly newsletter, a library catalogue and your first order form. In addition to the membership fee, a charge is made every time a tape is loaned out.

The library has to keep more than one copy of many of its titles. There are as many as 20 copies of some of the most popular cassettes.

Each cassette has a surprisingly short life. "Usually a cassette won't last much more than 15 or so lendings," says Alec. "By that time someone has creased the tape or it gets damaged in the post."

Each cassette is loaned for a two week period but in practice most are returned before the time expires. Just like a book library, every cassette has an accompanying card stamped with its return dates. Looking at these it is easy to see that the library stock seldom languishes on the shelves — tapes are often taken out and returned several times a month.

"When members return their programs we encourage them to give it a score. These assessments are then fed into my ZX81 and we compile a top 20 list of cassettes — based on how the members rate the programs rather than on the number of times it is taken out. The ones most in demand do not necessarily get the highest score."

Most members hire new tapes immediately after returning the old ones. This means over 50 returns every day. So the library now employs three people part-time, as well as Alec and his wife Erna.

"Nearly all our tapes are out on hire at any one time — if we have tapes on our shelves they tend to be the 1K ZX81 programs. At the moment adventures are going well and we get a lot of demand for Spectrum utilities."

To a software house the main headache a library throws up is that of illegal duplicating. Copying cassettes while they are out on hire is a very difficult problem to control.

"To start with I wasn't sure what sort of reactions we would get from software suppliers. I made one or two tentative enquiries before we set up the library and those software companies we spoke to seemed quite happy with our idea.

"Our service operates just like any other lending library — it's all perfectly legitimate. In fact, many public lending libraries now offer a music cassette lending facility — the local library in Liss certainly does. We are only doing the same with computer cassettes.

"At first all the money we earned went into building up our catalogue. We soon discovered there was a big demand for what we were doing. Luckily all my business experience has been in mail-order. For the last 16 years I have been the managing director of a photographic supplies mail order company.

"I knew roughly what we would be in for if the library turned out to be a success. If I had not been prepared we would certainly have been swamped — we are still getting 50 to 60 new members per week!

"We cater equally for ZX81 and Spectrum owners — in fact our membership is split right down the middle. Our range of ZX81 programs is greater simply because the machine has been around longer.

"We choose which tapes we stock. There is often more than one program that does much the same thing, particularly with utilities. Selection is made on the basis of manufacturer's literature and all our main suppliers keep us informed of new products.

"Of course there are a few suppliers who just don't want to know about software libraries. Some state on the cassette that it may not be hired out.

"We always make our intentions clear when ordering tapes for the library. Of all the companies we have contacted, only eight will not sell to us — and we respect that view. In such cases we simply do not have those programs in our library.

"All the tapes we hire out are bought from the manufacturer and many companies regard us as a good customer because of the quantities we buy. We could never stock all of the material from each company. If a member hires a tape from us and likes it then he or she may well go out and buy it for themselves. Alternatively, they may find that the programs from a particular company are very good and when they have a new product buy it.

Luckily all his business experience has been in mail order

"Obviously we discourage them from illegally making copies. Many of our suppliers send us special versions of their programs which auto-run and cannot be saved.

"It is a rule of membership that library cassettes must not be duplicated.

"It has been suggested that we should pay a royalty to software companies. A figure of 20 percent has been mentioned, which would be ludicrous — it would be more than the hire fee.

"The software houses get their cut anyway — both on the new tapes we buy and on the replacements bought later. I'm sure tapes wear out much more quickly than books do in public libraries." ■

Escape!

***John Scriven** finds out whether the latest Spectrum games are good enough to save you money in the arcades*

A friend of mine said last year that he'd recently bought a micro to save money. On enquiring how this might be achieved, I was surprised to discover that it was not to help with home finances, producing sales graphs or calculating odds on football pools. He had calculated that he spent £1.50 on video games every time he visited his local hostelry.

If he were to buy a ZX81, he could stay in and play arcade games and, in less than four months, he'd have saved himself the purchase price. Needless to say, he was back at the 20p slot within a month, disillusioned with the standard of the games he'd acquired.

Most were in Basic, very slow and could not compete with the colour and sound of the real thing. In the 18 months since this happened, micros have advanced considerably. The speed has been improved by the use of machine code in many games.

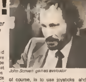
This review will consider how far the successor to the ZX81, the Spectrum, can emulate the original arcade games, or even surpass them.

There are several games based on the Pacman theme as well as one or two Invader look-alikes. It would be pleasant to find rather more innovation in game concepts, but it seems the great British arcade-playing public prefers tried and tested ideas and new games take time to catch on.

Mazeman from Abersoft is a maze pursuit game that involves eating dots while avoiding four little ghosts who pursue you. If you've recently eaten a power pill, the hunters become the hunted for about eight seconds. Tackling them gives you extra points. There is on-screen display of men left, screens eaten, individual score and hi-score. On the whole it is a competent high-speed version but the choice of cursor control keys for movement, although logical, does not make for ease of playing.

Spookyman from Abbex is similar in concept to *Mazeman* and does have the advantage of easier control keys. In fact any key in the top row of the keyboard moves your player up, the bottom row moves it downwards, and the middle two rows are divided in half for left and right movement. This means you can select which keys are most suitable for you.

The reason why this is preferable to the cursor keys is that the movement on the screen is related to the geographical positions on the keyboard. It is more suitable for high-speed action games. The answer


John Scriven, games evaluator

of course, is to use joysticks, and few games mention this facility.

This will doubtless change in 1983 when Sinclair, as well as Kempston, produce a joystick. *Spookyman* is very fast and does have a one- or two-player option, although all your turns have to be taken consecutively.

Spectres is the Bugbyte maze game and is similar to the two previous games. The graphics are more advanced and there is the entertaining story of Eddie the electrician trying to turn the lights on in a haunted house. But it is still a Pacman sheep in wolf's clothing. This is a fine product, but at £8, it is £3 more than the other two games, and as such, rather over-priced.

In its newly-announced collection of



software, Sinclair has included what appears to be yet another Pacman in the guise of *Hungry Horace*. It is soon clear, however, that a spark of originality lifts this game above run-of-the-mill maze games.

The maze has bridges and tunnels, an exit and an entrance through which an endearing little man appears. He has to be steered round, munching fruit that occasionally appears, while you avoid purple guards. If you reach what appears to be a bell, you can temporarily turn the tables on the guards and chase them. Should you negotiate the first maze successfully, there are three others that increase in complexity, the last one leading back to the first, but with an increase in difficulty. There is a sensible choice for movement keys and the sound of munching is very realistic.

This is one of the best Spectrum games and very addictive. It is noticeable that all the Sinclair cassettes produced in conjunction with Psion are easily loaded and well-written.

New Generation Software has managed to achieve a maze games (*Escape*) that is both original and entertaining. The maze appears viewed from an angle of 45°, giving a 3-D effect. Vertical paths are obvious, horizontal ones often obscured by hedges. Difficulty is selectable from 1 to 5, and the object is to find a hidden axe and use it to batter down the exit.

No problem, you may think, except that dinosaurs (the same number as the difficulty level) pursue you. The graphics are excellent, especially a horrifying pterodactyl from which it is almost impossible to escape. Top scores are recorded, as is the time taken. My one criticism is the familiar difficulty of using the cursor keys for control.

There are two versions of Asteroids, *Planetoids* from Sinclair/Psion, and *Meteoroids* from Softek. The original arcade game provided you with a small triangular spaceship in the centre of the screen. Two buttons controlled rotation, and two more thrust and lasers. A panic button could hyperspace you to another part of the video universe.

The main enemy consisted of large chunks of interplanetary detritus that broke up until they were eventually vaporised. Additional excitement was provided by enemy saucers that shot at you. Avoiding this collection while destroying it was a challenge, but the graphics belonged to an earlier generation of arcade money-spinners.

Planetoids copies the original faithfully, but uses the user-defined graphics facility to produce a much more life-like ship. Unfortunately the movement is not realistic, being both jerky and too easy to control. The original needed great skill to learn to use reverse thrust to prevent the ship careering off screen.

Softek's version suffers from a similar disregard for the laws of physics and has a simpler spacecraft. The meteoroids, however, are very solid in appearance and the game is more involved than Sinclair's, having shield and movement for protection. There is also the option to temporarily halt the game while you do the washing-up, the gardening or your homework without destroying your brilliant score. These are two versions of a rather dated game. Softek just wins on points. Sinclair's version does have a short game called *Missile* on the reverse which probably makes them of equal value.

Next come the obligatory versions of Space Invaders (yawn!), one from Sinclair/Psion, *Space Raiders*, and one from Quicksilva, *Space Intruders*. Both include banks of invaders, laser cannons and buildings to shelter beneath. Sinclair's game has better graphics but is painfully slow. Quicksilva's version produces neat little invaders but rather simplistic ground shelters that disappear in big chunks and



cannot be used to fire through This was a favourite trick on the arcade version It is however fast enough to keep you awake during play, which is more than can be said for Space Invaders

Another game that involves protecting a base from falling objects is *Rox III* from Llamasoft. You will need plenty of practice with this game to become proficient at destroying meteors as they crash near your moon base Unfortunately, the advertising calls these Cruise missiles, which is factually incorrect — Lunar ballistic missiles would be more accurate — and in any case probably offends the not inconsiderable number of unilateralists in this country The game itself is exciting and well written, as well as being good value at only £2 95

Two games that push the potential of the Spectrum to its limits both originate from Silversoft Perhaps "originate is not the best word as they are both extremely good copies of complex arcade games *Ground Attack* is a version of Scramble, in which you negotiate a tortuous tunnel system, bombing fuel dumps and shooting at rockets There are controls for up down and sideways movement as well as bombs and laser buttons It is a test of real dexterity to cope with the later stages of the tunnel Good value at £5 95

Silversoft's *Orbiter* seems to have reached the limits of Spectrum graphics It reproduces almost all the features of *Defender* and is only slightly slower The attack waves are all there, complete with little men, mutants aliens cluster bombs and the ability to fly to the left or the right There is also the small radar screen at the top to show what sort of nasties are approaching

I even found that using *Orbiter* for a few days improved my score on the arcade version My small criticism is that there was no provision for a table of best scores I'm sure that a great incentive to play arcade games is the ability to flash your name to all and sundry when you reach the top ten Notwithstanding this, it's well worth £5 95

There are two recently released games that attempt to boldly go where no arcade games have gone before *Cosmos* from Abbex puts you at the controls of a spacecraft defending a convoy from the ravages of marauding aliens and the odd meteoroid A radar screen in the corner of the main screen shows your relative position while the rest of the screen is taken up with the view from the cockpit

Rushing into the game without studying the clear on screen instructions caused me to blast away at my own convoy thus scoring the minimum points in about 10 seconds flat The next time I took more care to explore the possibilities of the game Although novel in concept, I felt it lacked the speed one expects from this type of game It is interesting enough however to find a place in many people's collections

Time-Gate from Quicksilva is described as a '4-D adventure' It is the most complicated cassette that is reviewed here and contains not only the program of the game but also a short training program explaining the scenario and the use of the controls This is obligatory viewing otherwise you will not have the faintest idea what is happening

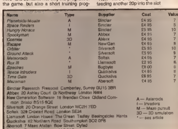
The sleeve notes on the cassette contain the traditional Quicksilva Sci-Fi story just to put you in the right mood To be honest, I would have preferred a list of the large number of control keys Meanwhile the story so far ... this end of the universe has been invaded for several millennia by reptilian thugs In order to eradicate them it is necessary to discover time-gates that lead you back to the time when they first appeared Destroy them before they breed and mankind is saved for posterity, or at least until you run the game again

The screen display consists of the view from the front of the craft, a galactic co-ordinate chart, and a target computer Steering and fire controls are simplified by a keyboard template that slips over a section of the keys There is provision for use of a joystick It is possible to change speed to jump to another sector of the universe and to land on a planet to refuel

This graphic sequence is particularly striking, as is the 3D effect as you battle it out with assorted aliens In spite of the excellent graphics and use of screen to show spacecraft condition, I was not alone in finding the initial excitement beginning to pall as I waded back through time Waiting to catch up with fleeing aliens was irritating

Although I have reservations *Time-Gate* is a complex visually superb game that is to be commended on its novelty

All the games here show how far home computers have progressed over the past 18 months I wanted to show my friend with the ZX81 what he was missing but he wasn't at home, I discovered him later in the corner of my local scampering up video trees as he played "Donkey Kong" "Now here's a real game he said feeding another 20p into the slot ■



Name	Type	Supplier	Cost	Value
Planetoids/Missile	A	Sinclair	£4 95	7
Space Raiders	I	Sinclair	£4 95	5
Hungry Horace	M	Sinclair	£5 95	10
Spookyman	M	Abbex	£4 95	7
Cosmos	3D	Abbex	£4 95	7
Escape	M	New/Gen	£4 95	9
Orbiter	*	Silversoft	£5 95	10
Ground Attack	*	Silversoft	£5 95	9
Meteoroids	A	Softek	£4 95	7
Rox III	*	Llamasoft	£2 95	8
Spectres	M	Bugbyte	£8 00	6
Space Intruders	I	Quicksilva	£4 95	9
Time Gate	3D	Quicksilva	£6 95	7
Mazeman	M	Abersoft	£4 95	7

Sinclair Research Freepost Camberley, Surrey GU15 3BR
Abbex 20 Ashley Court Gt Northway London NW4
New Generation Software 11 Brendan Close Oldland Common Bristol BS15 6QE
Silversoft 20 Orange Street London WC2H 7ED
Softek 329 Croxted Road London SE24
Llamasoft Lindon House The Green Tadley Basingstoke Hants
Quicksilva 92 Northern Road Southampton SO2 0PB
Abersoft 7 Maes Afallen Bow Street Dyfed
Bugbyte Freepost Liverpool L3 3AB

A — Asteroids
I — Invaders
M — Maze pursuit
3D — 3D simulation
* — see article

Open Forum is for you to publish your programs and ideas. Take care that the listings you send in are all bug-free. Your documentation should start with a general description of the program and what it does and then give some detail of how the program is constructed. We will pay the *Program of the Week* double our new fee of £6 for each program published.

Race Maze

on ZX81

This is a games program, in which you are challenged to race your car through a complicated maze in the shortest possible time. If you are unlucky and crash, five seconds are added to your final time. At the very beginning of the program the instructions are printed out then the screen is cleared and the maze is printed out in fast-mode.

The movement of the car and the checking to see whether you have crashed is all done using *Peek* and *Poke*. Then the rest of the program is made up of the subroutines: one for crashing and one for printing out the end time and crashes.

```
  6 REM  MAZE
  6 LET B=11
  7 LET M=1
  8 LET Z=0
  9 LET A8=[illegible]
 10 LET C=173
 11 FAST
 12 PRINT AT 5,5;"THIS IS [illegible]"
 13 PRINT AT 6,0;"STEER YOUR CAR THROUGH THE MAZE"
 14 PRINT AT 7,1;"USEING ARROW KEYS 5,8,[illegible]"
 15 PRINT AT 8,1;"YOU ARE SCORED IN THE TIME IT TAKES YOU TO COVER THE COURSE"
 16 PRINT AT 10,1;"5 SECS ARE ADDED TO YOUR TIME EACH TIME YOU CRASH          PRESS N/L TO START"
 45 INPUT T$
 46 CLS
 47 LET S=0
 50 FOR A=1 TO 21
 55 PRINT "[illegible]"
 60 NEXT A
 75 LET S=21
 80 FOR A=1 TO 21
 85 PRINT AT S,31;"[illegible]"
 90 LET S=S-1
100 NEXT A
110 PRINT AT 0,0;"[illegible]"
111 PRINT AT 21,0;"[illegible]"
115 LET F=1
120 FOR D=1 TO 20
130 PRINT AT F,1;"[illegible]"
140 LET F=F+1
150 NEXT D
160 LET K=7
170 FOR J=1 TO 8
175 PRINT AT 14,20;"[illegible]"
176 PRINT AT 8,20;"[illegible]"
177 PRINT AT 11,18;"[illegible]"
180 PRINT AT K,12;"[illegible]"
189 LET K=K+1
191 FOR N=8 TO 13
192 PRINT AT N,13;"[illegible]"
193 NEXT K
200 NEXT J
201 FOR O=9 TO 12
202 PRINT AT O,14;"[illegible]"
203 NEXT O
205 PRINT AT 10,18;"[illegible]"
206 PRINT AT 11,15;"[illegible]"
207 PRINT AT 8,13;"[illegible]"
208 PRINT AT 13,13;"[illegible]"
209 PRINT AT 12,14;"[illegible]"
210 PRINT AT 1,2;"[illegible]";AT 1,8;"[illegible]";AT 2,26;"[illegible]";AT 2,15;"[illegible]"
215 LET W=3
220 FOR L=1 TO 14
230 PRINT AT W,2;"[illegible]"
235 LET W=W+1
240 NEXT L
250 PRINT AT 20,8;"[illegible]"
255 LET R=2
260 FOR E=1 TO 13
261 PRINT AT 1,30;"[illegible]"
262 PRINT AT 1,26;"[illegible]"
270 PRINT AT R,8;"[illegible]"
275 LET R=R+1
277 NEXT E
280 PRINT AT 20,0;"[illegible]"
290 PRINT AT 16,6;"[illegible]";AT 17,8;"[illegible]"
295 PRINT AT 18,8;"[illegible]"
300 PRINT AT 19,25;"[illegible]"
310 PRINT AT 3,6;"[illegible]"
320 PRINT AT 8,16;"[illegible]"
330 PRINT AT 16,19;"[illegible]"
340 PRINT AT 15,19;"[illegible]"
350 LET Y=3
360 FOR E=1 TO 10
365 PRINT AT Y,25;"[illegible]"
370 LET Y=Y+1
375 NEXT E
377 PRINT AT 7,28;"[illegible]";AT 8,28;"[illegible]"
380 LET O=8
381 FOR N=1 TO 10
382 PRINT AT O,8;"[illegible]"
383 LET O=O+1
384 NEXT N
385 LET O=6
386 FOR K=1 TO 10
387 PRINT AT 6,18;"[illegible]"
388 LET O=O+1
389 NEXT K
390 PRINT AT 9,19;"[illegible]"
395 LET T=4
400 FOR N=1 TO 10
401 PRINT AT T,21;"[illegible]"
402 LET T=T+1
403 NEXT N
404 PRINT AT 10,21;"[illegible]";AT 13,[illegible]
406 LET U=4
406 FOR N=1 TO 14
407 PRINT AT U,23;"[illegible]"
408 LET U=U+1
409 NEXT N
410 PRINT AT 2,23;"[illegible]";AT 3,23;"[illegible]"
411 LET W=2
412 FOR N=1 TO 16
413 PRINT AT W,25;"[illegible]"
414 LET W=W+1
415 NEXT N
416 PRINT AT 4,25;"[illegible]";AT 5,25;"[illegible]"
417 SLOW
420 LET PI=28076
425 POKE PI,138
426 LET BI=PI
430 LET PI=PI+(INKEY$="8")-(INKEY$="5")+33*((INKEY$="6")-(INKEY$="7"))
433 IF PEEK PI=137 THEN GOSUB 2000
435 IF PEEK PI=136 THEN GOTO 3000
440 IF PEEK PI=8 THEN GOSUB 2000
445 IF PEEK PI=3 THEN LET PI=[illegible]
446 IF PEEK PI=8 THEN LET PI=[illegible]
447 IF PEEK PI=131 THEN LET PI=[illegible]
500 IF INKEY$="6" OR INKEY$="[illegible]" THEN LET C=174
501 IF INKEY$="5" OR INKEY$="[illegible]" THEN LET C=173
505 POKE PI,C
506 LET Z=Z+1
700 GOTO 425
999 STOP
1000 PRINT [illegible]
1001 RETURN
1002 STOP
2000 LET Z=Z+8
2010 LET E=E+1
2020 POKE PI,25
2025 PAUSE 30
2030 RETURN
3000 LET W=1
3001 FOR R=1 TO 20
3002 PRINT AT W,30;"[illegible]"
3003 LET W=W+1
3004 NEXT R
3010 FOR N=7 TO 14
3011 PRINT AT N,12;"[illegible]"
3012 NEXT N
3020 PRINT AT 7,12;"[illegible]"
3030 PRINT AT 8,14;"[illegible]"
3040 PRINT AT 10,14;"[illegible]"
3050 PRINT AT 11,14;"[illegible]"
3060 PRINT AT 13,12;"[illegible]",Z
3070 FOR P=1 TO 10
3080 NEXT P
3085 PRINT AT 10,14;"[illegible]"
3086 PRINT AT 11,14;"OVER"
3087 PRINT AT 13,12;"TIME=";Z,
4000 FOR G=1 TO 10
4001 NEXT G
4005 GOTO 3050
5000 FOR N=28030 TO 30000
5010 IF PEEK N=8 THEN POKE N,128
5020 NEXT N
6010 CLS
6020 FOR N=1 TO 4
6030 FOR J=1 TO 20
6040 PRINT "BEST TIME 80 PAR"
6045 SCROLL
6046 NEXT J
6047 NEXT N
6050 LET P=Z
6055 RETURN
```

Race Maze

by Mark Greer

Hol Sin

on Spectrum

This not so ancient Chinese game will run happily on your equally not so ancient ZX Spectrum. It will require oriental patience and much eastern ingenuity.

There are no difficulty levels to choose between because there is only one level — difficult. The instructions are included in the listing and are very comprehensive.

There are no aliens to shoot down, no mazes to get through, nor time limits to beat: all you have to beat is your own ability to think logically.

There is an old Chinese proverb which says the man who can be defeated is the man who does not try.

Program notes

1 to 175	Setting up arrays and instructions
188	Clears b$
190 to 210	Sets up a random board
230 to 260	Inputs and checks move
265 to 275	Makes the move
280 to 285	Checks for completion
290 to 315	Displays score, asks for another game
335 to 360	Special move
Subroutine 1130	prints reference board
Subroutine 1230	prints the up-to-date working board

```
  5 LET s=0: LET e1=0: LET S=1
  8 DIM b(16): DIM b$(16)
 10 INPUT "Do you want instructions ? (y/n) ";a$
 20 BORDER 1: CLS
 25 POKE 23609,50
 30 GO SUB 1130
 45 IF a$="n" THEN GO TO 180
 55 PRINT AT 0,0
 60 PRINT INK 7; PAPER 1;"The board positions are randomly occupied with the letters 'A' TO 'P'. The object of the game is to rearrange the letters into order by ROTATING a group of four letters CLOCKWISE on position";: INK 9: PAPER 5;"You specify the upper left position of the four you wish to rotate"
 65 PRINT AT 21,3; PAPER 7; INK 2; BRIGHT 1;"Press any key to continue": PAUSE 9000: PRINT AT 4,14; PAPER 7; INK 2;"Valid moves are
 70 PRINT PAPER 7; INK 1;AT 6,1[illegible]"1 2 3 5 6";AT 6,16;"[illegible] 10 11"
 75 PRINT OVER 1; FLASH 1;AT 3,1;"[illegible]";AT 3,4;"[illegible]";AT 3,[illegible];AT 4,1;"[illegible]";AT 4,4;"[illegible]";AT 4,[illegible];AT 5,1;"[illegible]";AT 5,4;"[illegible]";AT 5,5;"[illegible]";AT 5,5;"[illegible]"
 80 FOR f=0 TO 21: PRINT PAPER 1;AT f,0;,: NEXT f
 85 PRINT AT 21,3; PAPER 7; INK 2; BRIGHT 1;"Press any key to continue": PAUSE 9000: CLS : BORDER 7
 90 PRINT PAPER 1; INK 7;"If the board looked  , like this [illegible]
 95 FOR i=1 TO 16: LET b$(i)=CHR$ (i+64): NEXT i
100 LET b$(6)=[illegible]: LET b$(3)="[illegible]": LET b$(6)="b": LET b$(7)="f"
105 GO SUB 300
110 PRINT PAPER 6; OVER 1; BRIGHT 1;AT 3,3;"[illegible]";AT 3,5;"[illegible]";AT 4,3;"[illegible]";AT 4,5;"[illegible]": PAUSE 150
115 PRINT PAPER 5;AT 3,18; INK [illegible]"and you";AT 4,18;"rotated";AT 5,18;"position";AT 6,18;"2": PRINT OVER 1; FLASH 1;AT 3,3;"[illegible]": FOR f=0 TO 700: NEXT f
118 PRINT PAPER 1; INK 7;AT 9,10;"The board";AT 10,10;"becomes"
120 FOR i=0 TO 7: LET b$(i)=CHR$ (i+64): NEXT i
130 GO SUB 1030
135 LET S=1: PRINT PAPER 1; INK 7;AT 16,10;"and you";AT 17,10;"would WIN"
150 PRINT AT 21,3; PAPER 7; INK 2; BRIGHT 1;"Press any key to continue": PAUSE 9000: CLS : BORDER 7
155 CLS
160 PRINT PAPER 1; INK 7;"AND you also have one SPECIAL move per game which you may or may not need.      The special move allows you to swap two ADJACENT letters, in a ROW."
```

PROGRAM OF THE WEEK

```
160 PRINT PAPER 1, INK 7, To
ease this move input -1 as yo
ur move, and you will be ask
ed for the two positions of the
letters you wish to exchange
170 PRINT PRINT PAPER 5 INK
0 REMEMBER ONLY ONE SPECIAL MOV
E PER GAME
PRINT PRINT PAPER 1, I
NK 7, To give up at any time inp
ut 0 PAUSE 100
175 PRINT AT 21 0 PAPER 7 INK
2 BRIGHT 1 Press any key to c
ontinue PAUSE 0: CLS BORDER
5 GO SUB 1130
180 PRINT PAPER 1, INK 7,AT 10,
0, O.K I M THINKING UP A HARD O
NE FOR YOU

185 FOR I=1 TO 16 LET B$(I)=
NEXT I
190 FOR I=1 TO 16
195 LET T$=CHR$ (INT (RND*15+65
))
200 FOR J=1 TO I IF B$(J)=T$ T
HEN GO TO 195
205 NEXT J
210 LET B$(I)=T$ NEXT I
215 PAUSE 150
220 LET M=0 LET S=0 GO SUB 12
30
225 PRINT PAPER 1 INK 7 AT 18,
0 MOVES TAKEN SO FAR =
M,

230 PRINT AT 12,0,
235 INPUT PAPER 4 TRY 0 Possit
ion to ROTATE 10
240 FOR h=1 TO LEN i$ IF CODE
i$(h)<48 OR CODE i$(h)>57 THEN G
O TO 235
245 NEXT h LET i=VAL i$
250 IF I=0 THEN GO TO 380
255 IF i=-1 THEN GO TO 335
260 IF i<4 OR i 8 OR i>=12 THEN
PRINT AT 19 0 PAPER 1 INK 7;"
ILLEGAL MOVE- TRY AGAIN
" GO TO 235
265 LET M=M+1 LET T$=B$(I) LE
T B$(I)=B$(I+4) LET B$(I+4)=B$(
I+5) LET B$(I+5)=B$(I+1) LET B
$(I+1)=T$
270 GO SUB 1230
275 PRINT PAPER 7 AT 19,0,"
PAPER 1 INK 7,AT 18,0," MOVES
TAKEN SO FAR = ";M
280 FOR I=1 TO 16: IF CHR$ (I+6
4)<>B$(I) THEN GO TO 230
285 NEXT I
290 PRINT PAPER 1, INK 7,AT 19,
0, YOU ORDERED THE BOARD IN
,M MOVES

295 LET M1=M1+M LET G=G+1
300 INPUT Play again (y/n) ?",
a$
305 IF a$(1)="y" THEN GO TO 10
310 FOR q=1 TO 60 BEEP .005,q
NEXT q
315 PRINT PAPER 1, INK 7,AT 18,
0 you played G games and h
ave an average score of M1/G,
moves/game STOP
320 PRINT PAPER 7 INK 0 AT 0 0
325 FOR i=1 TO 16 STEP 4
330 PRINT PAPER 0 INK 0 b$
(i) B$(i+1), B$(i+2)
B$(i+3) NEXT i PLOT 1 152 DRA
U 83 0 DRAW 0 -33 DRAW -83,0
DRAW 0,33 RETURN
335 IF s>=1 THEN GO TO 375
340 PRINT AT 21 0,"
" INPUT PAPER 4, INK
0, First position ? ,x$
345 FOR h=1 TO LEN x$ IF CODE
x$(h)<48 OR CODE x$(h)>57 THEN G
O TO 340
350 NEXT h LET x=VAL x$
355 INPUT PAPER 4, INK 0 Secon
d position ? y$
360 FOR h=1 TO LEN y$ IF CODE
y$(h)<48 OR CODE y$(h)>57 THEN G
O TO 355
365 NEXT h LET y=VAL y$
370 IF x<>y+1 AND x<>y-1 OR x>=
17 OR y>=17 THEN PRINT AT 19,0,
PAPER 1, INK 7, ILLEGAL MOVE
TRY AGAIN PAUSE 100 PRI
NT AT 19 0 "
" GO TO 340
375 LET s=s+1 IF s>1 THEN PRIN
T AT 21,0, INK 2, BRIGHT 1 ONL
Y ONE SPECIAL MOVE PER GAME P
AUSE 100 PRINT AT 21,0
" GO T
O 235
380 LET i$=b$(x) LET b$(x)=b$(
y) LET b$(y)=i$: GO TO 270
1130 FOR i=1 TO 16 LET b(i)=i
NEXT i PRINT ,,,
1135 FOR i=1 TO 13 STEP 4 PRINT
PAPER 0 TAB 1,b(i),TAB 4,b(i+1)
TAB 7,b(i+2),TAB 11,b(i+3) NEX
T i
1140 PRINT PAPER 0,AT 3 10," ,A
T 4,10 PLOT 1,150 DRAW 104
,0 DRAW 0,-33 DRAW -104,0 DRA
W 0,33
1190 RETURN
1230 PRINT PAPER 7 INK 0 AT 12,
0
1235 FOR i=1 TO 16 STEP 4
1240 PRINT PAPER 0 INK 0 b$
(i), b$(i+1) b$(i+2)
b$(i+3) NEXT i PLOT 0 39 DRAW
84 0 DRAW 0,33 DRAW -84,0 DR
AW 0 -33 RETURN
```

Hoi Sin
by D Wieckowski

Screen scrolls

on ZX81

The ZX81's *Scroll* and *CLS* routines are very slow, and this can be frustrating when you are trying to write fast-action games in Basic. Also the lack of lateral scrolls and a downward scroll can be quite maddening.

To get over this I have written five short machine code routines that are totally relocatable in Rem.

To work through, these routines need at least 3¼K of Rem.

The first listing is ready for treatment by Bug Byte's ZXAS program — for those fortunate enough to own one. I have also given a simple Hex Loader program and a Hex Dump of the machine code, for programmers without the ZXAS program. To relocate the program just change the addresses set in line 20.

To enter the machine code type in the simple Hex loader and *Run*. Now key in the complete Hex Dump. If you make a mistake type S and *Run* again. When you have finished delete lines 10-130. The machine code is now held in line 2 and is ready for use.

To enter the machine code with an assembler enter line 2 as in the simple Hex loader and type in the Mnemonics. Now GOTO 9000 and then delete lines 3 9060.

If you do relocate the program starting at address X then the call up points are as follows

Scroll Up	= X
Scroll Down	= x + 27
Scroll Right	= x + 56
Scroll Left	− x + 95
CLS	= x + 118

The routines only scroll the first 22 lines leaving the bottom two free for scores/times to be *Poked* in and left unchanged after using the *Scrolls* or *CLS* routine.

Only one line is moved at once like the ZX81's scroll command does. It also blanks the line that is left stationary by the routines.

The Assembler Listing

Line 2 holds the machine code

Line 10 opens the assembler file

Lines 30-150 scroll the screen up by using the instruction LDIR. The routine uses registers DE HL and BC

Lines 170 330 scroll the screen down by using LDDR. It uses registers BC, HL and DE

Lines 350-530 move the screen to the right. It uses LDDR in a different configuration to before. The registers used are A, HL, DE and BC

Lines 550-720 move the screen left.

The routine uses the instruction LDIR to do all the moving. Registers used are A, HL, DE and BC

Lines 740-890 clear the screen. By adding lines

```
815 SET 7 A        CB FF
820 LD (HL),A      77
```

the CLS routine can become a screen invert routine. But to do this line 2 needs one extra "X" and the RESET PRINT POSITION routine moves up by one byte but the label system on the Assembler automatically corrects this. The registers used are HL, BC and A.

Lines 910-980 are the reset print position routine. This is required by all routines otherwise some important system variables will be set incorrectly upon returning to Basic

Line 990 closes the assembler file

Lines 9000-9060 are the assembler control program. Line 9010 should be changed if you wish to relocate the program

I have also written a very simple demonstration program just to show one way of using the routines but they have limitless capabilities

These routines are very flexible so have a go at adapting them to do different jobs, such as blanking the unscrolled line with a different character. It is easier to manipulate the routines if they are stored above a lowered Ramtop

```
HEX DUMP OF SCREEN SCROLLS

16514   ED 5B 0C 40 2A 21 21 00
16522   13 01 B5 02 C5 ED B0 C1
16530   E1 09 06 20 23 36 00 10
16538   FB 18 AA 01 D6 02 2A 0C
16546   40 ED B5 54 5D 01 B5 0A
16554   21 ED 09 ED 08 05 20 E1
16562   23 3E 00 33 10 FB 18 4E
16570   2F 05 1A 2A 0C 40 11 17
16578   00 10 54 5D 2B C5 01 1F
16586   00 ED 0B 1E 16 1B 26 10
16594   C1 10 F2 18 31 AF 06 15
16602   2A 0C 40 23 54 5D 23 C5
16610   01 1F 00 ED 00 12 13 13
16618   D3 20 C1 10 F2 18 17 2A
16626   0C 40 23 01 D6 02 7E FE
16634   76 3E 02 3E 0B 23 0B 7F
16642   0B 20 F3 06 20 F0 2A 0C
16650   40 23 22 0E 42 21 3B 40
16658   36 21 [illegible] 3E 1[illegible] [illegible] [illegible] [illegible]

SIMPLE HEX LOADER

  2 REM ( 180 "X"S )
 10 FAST
 20 FOR A=16514 TO 16683
 30 SCROLL
 40 PRINT A, "-",
 50 SLOW
 60 INPUT S$
 70 FAST
 80 IF S$=" " THEN GOTO 50
 90 IF S$="S" THEN STOP
100 PRINT S$
110 LET B=CODE S$*16+CODE S$(2)
120 POKE A,B
130 NEXT A

MACHINE CODE CALL ADDRESSES

16514=SCROLL UP
16541=SCROLL DOWN
16570=SCROLL RIGHT
16609=SCROLL LEFT
16624=CLS

DEMONSTRATION PROGRAM

2 REM FOR/UR [illegible]
[illegible]
[illegible]

3 LET UP=16514
4 LET DOWN=16541
5 LET RIGHT=16570
6 LET LEFT=16609
```

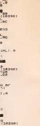
Continued on page 18

Continued from page 15

```
   7 LET CLS=16824
   8 LET CLSCHAR=16838
  10 SLOW
  20 GOSUB 1000
  30 FOR A=1 TO 22
  40 RAND USR UP
  50 NEXT A
  60 GOSUB 1000
  70 FOR A=1 TO 22
  80 RAND USR DOWN
  90 NEXT A
 100 GOSUB 1000
 110 FOR A=1 TO 32
 120 RAND USR RIGHT
 130 NEXT A
 140 GOSUB 1000
 150 FOR A=1 TO 32
 160 RAND USR LEFT
 170 NEXT A
 180 FOR A=0 TO 63
 190 POKE CLSCHAR,A
 200 RAND USR CLS
 210 NEXT A
 220 FOR A=128 TO 191
 230 POKE CLSCHAR,A
 240 RAND USR CLS
 250 NEXT A
 260 POKE CLSCHAR,0
 270 RAND USR CLS
 280 PRINT AT 11,12,"FINISHED"
 290 STOP
1000 POKE CLSCHAR,128
1010 RAND USR CLS
1020 PRINT AT 0,0, TL"
1030 PRINT AT 0,30 "TR"
1040 PRINT AT 21,0, "BL"
1050 PRINT AT 21 30, "BR"
1060 RETURN
```

ASSEMBLER LISTING

```
   2 REM GOSUB 7ERNDSTR$ 80 ,[redacted]
VAL GOSUB [redacted] LPRINT [redacted]
[redacted] LIST EERND7STR ANDSTANDSTR TR
N
   3 REM
   5 REM SCREEN SCROLLS READY
FOR ASSEMBLING BY BUG-BYTES ZXAS
  10 REM L
  15 REM [redacted]
  20 REM [redacted]
  30 REM LD DE (16396)
```



```
  38 REM PUSH DE
  40 REM LD HL 33
  50 REM ADD HL,DE
  60 REM LD BC,693
  65 REM PUSH BC
  70 REM LDIR
  80 REM POP BC
  90 REM POP HL
 100 REM ADD HL,BC
 110 REM LD B 32
 120 REM L8INC HL
 130 REM LD (HL),0
 140 REM DJNZ L8
 150 REM JR L8
 160 REM [redacted]
 170 REM LD BC 728
 180 REM LD HL (16396)
 190 REM PUSH HL
 200 REM ADD HL,BC
 210 REM LD D H
 220 REM LD E L
 225 REM LD BC,693
 230 REM POP HL
 240 REM PUSH HL
 250 REM ADD HL,BC
 260 REM LDDR
 270 REM LD B 30
 280 REM POP HL
 290 REM INC HL
 300 REM L1LD (HL) 0
 310 REM INC HL
 320 REM DJNZ L1
 330 REM JR 6
 340 REM [redacted]
 350 REM XOR A
 360 REM LD B 22
 370 REM LD HL (16396)
 380 REM LD DE 728
 390 REM ADD HL,DE
 400 REM LD D H
 410 REM LD E L
 420 REM DEC HL
 430 REM L3PUSH BC
 440 REM LD BC 31
 450 REM LDDR
 460 REM LD (DE),A
 470 REM DEC DE
 480 REM DEC DE
 490 REM DEC HL
 500 REM DEC HL
 510 REM POP BC
 520 REM DJNZ L3
 530 REM JR L6
 540 REM [redacted]
 550 REM XOR A
 560 REM LD B 22
 570 REM LD HL,(16396)
 580 REM INC HL
 590 REM LD D H
 600 REM LD E L
```



```
 610 REM INC HL
 620 REM L3PUSH BC
 630 REM LD BC 31
 640 REM LDIR
 650 REM LD (DE),A
 660 REM INC DE
 670 REM INC DE
 680 REM INC HL
 730 REM [redacted]
 740 REM LD HL (16396)
 760 REM LD BC 726
 790 REM L5LD A (HL)
 800 REM CP 118
 810 REM JR Z,L4
 820 REM LD (HL) 0
 830 REM L4INC HL
 840 REM DEC BC
 850 REM XOR A
 860 REM CP C
 870 REM JR NZ L5
 880 REM CP B
 890 REM JR NZ,L5
 900 REM [redacted]
 910 REM L6LD HL,(16 ...
 920 REM INC L
 930 REM LD (16396) HL
 940 REM LD HL 16441
 950 REM LD (HL) 33
 960 REM INC HL
 970 REM LD (HL) 24
 980 REM RET
 990 REM )
1000 REM FOR A 16K ZX-81
1010 REM
1020 REM WRITTEN BY K A STREATER
             11/7/82
1030 REM
9000 FAST
9010 LET Z22=16514
9020 POKE 32641 INT (Z22/256)
9030 POKE 32640,Z22-256*INT (Z22
/256))
9040 RAND USR 32655
9050 PRINT AT 21,0, ERROR ,PEEK
 32651
9060 SLOW
```

Screen scrolls
by K A Streater

Calendar for 1983

on BBC Micro

This program will draw a 1983 calendar. The variables used are:

D$(J)	=	Day of the week
Y$	=	Year for the calendar (1983)
M	=	Number of days in the month
M$	=	Name of the current month+Y$

L$ and LL$ are ruling lines

S$	=	Two spaces
T$	=	20 spaces
U$	=	Underscore the headings
L	=	Length of each heading
T	=	End position of a heading
J	=	1st January / loop variable
D	=	Days in month / loop variable

To run the program.

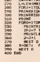
1 Type RUN, then adjust position of paper before switching on the printer to set the TOP OF FORM position. Press 'RETURN.
2 At the end of a month the printout will stop to allow you to adjust the paper position or insert a fresh sheet of paper.
3 Press the SPACE BAR to continue printing to the year's end.
4 Rulings can be changed by duplicating line 350 and inserting the extra L$ rulings at new line 335 so giving more space for each day's entries. A dummy GET statement after the J = 1 on line 340 will halt printing at the end of each week.
5 January 1st, 1983, is Saturday, the seventh day, hence J = 7 on line 220. For any other year the value for J must be reset and also Y$ in line 50. On leap years alter February 28 to 29 on line 150.

On the BBC Micro add line 90 VDU 1 27 1 65 1 10 to change the line spacing to 10/72 inch on the Epsom 80 printer and give a 10.5 inches page length for 31 days.

For many machines unpredictable results can be avoided by switching on the printer and entering all the printer instructions in direct mode from the keyboard first. These can be checked with a dummy run before the Basic program is loaded.

The string-forming routine on lines 260 to 300 is compatible with all Basics and will be found useful as a subroutine in other programs besides this one. M$ is left padded with spaces to ensure good centring whereas the dates are left-justified to a standard format in line 330. A double line is ruled at the end of each week in line 340.

```
 10 REM-------------------------------
 20 REM     CALENDAR FOR 1983
 30 REM-------------------------------
 40 DIMD$(7)
 50 S$="  ":Y$=" 1983"
 60 T$="                    "
 70 L$="----------------------------------------"
 80 LL$="========================================"
 90 :
100 DATASUNDAY,MONDAY,TUESDAY,WEDNESDAY
110 DATATHURSDAY,FRIDAY,SATURDAY
120 REM----------------------------
130 REM SET FEBRUARY,29 ON LEAP YRS.
140 REM----------------------------
150 DATAJANUARY,31,FEBRUARY,28
160 DATAMARCH,31,APRIL,30,MAY,31
170 DATAJUNE,30,JULY,31,AUGUST,31
180 DATASEPTEMBER,30,OCTOBER,31
190 DATANOVEMBER,30,DECEMBER,31
200 FORJ=1TO7:READD$(J):NEXT
210 REM-----------------------------
220 J=7:  REM Saturday = 1st January
230 REM ALTER J TO SUIT ANOTHER YEAR
```



```
240 REM--------------------------------
250 FORK=1TO12:READM$,M
260   M$=M$+Y$: U$=""
270   L=LEN(M$):T=L/2+20
280   FORI=1TOL:U$=U$+"=":NEXT
290   PRINTRIGHT$(T$+M$,T)
300   PRINTRIGHT$(T$+U$,T)
310   PRINT
320   FORD=1TOM
330     PRINTRIGHT$(S$+STR$(D),3);S$;D$(J)
340     IFJ=7 THEN PRINTLL$: J=1: GOTO 370
350     PRINTL$
360     J=J+1
370     NEXT D
380   X=GET:  REM  DUMMY INPUT PAUSE
390   NEXT K
400 END
```

Calendar for 1983
by L Hurst

Trace

on Vic20

This program is an analogue display for Vic20 with 3K Super Expender which uses a twin moving trace resembling that used in electrocardiographs oscilloscopes etc The various parameters can be quickly altered to suit any particular application I reckon this program could be of immense value to hobbyists and experimenters for monitoring and displaying various inputs from external equipment

The inputs are made through the control port of the Vic using the two paddle inputs The program as I ve supplied it runs as fast as possible (fastest trace scan) but should the user need a more rapid trace he can dispense with the Vertical numerical column or alternatively, increase the increments in lines 50 52 and 55.

The "unaffected" position of the two traces can be changed by altering the plussed-on values in lines 28 and 30 The traces automatically renew after each scan using line 56 Sound could be added to give a signal if the traces or just one trace, perhaps reaches a certain position, to sound an alarm

```
1 REM TWIN-TRACE DISPLAY
2 REM R.BARTON.
3 A=50
4 GRAPHIC2
6 COLOR0,3,1,1
10 CHAR0,0,"9" CHAR1,0,"8" CHAR2,0,"7" CHAR3,0,"6" CHAR4,0,"5"
11 CHAR5,0,"4" CHAR6,0,"3" CHAR7,0,"2" CHAR8,0,"1" CHAR9,0,"0"
12 CHAR10,0,"9" CHAR11,0,"8" CHAR12,0,"7" CHAR13,0,"6" CHAR14,0,"5"
13 CHAR15,0,"4" CHAR16,0,"3' CHAR17,0,"2" CHAR18,0,"1" CHAR19,0,"0"
20 Y1=PEEK(36072)+35
30 Y2=PEEK(36073)+560
50 DRAW2,A,Y1TOA+30,Y1
52 DRAW2,A,Y2TOA+30.Y2
55 A=A+30
56 IFA>=1020THEN SCHCLR A=50
100 GOTO10
```

Trace
by Richard Barton

Screen store

on Spectrum

This program is based on a very short machine code routine stored above Ramtoo which will load one of up to five screens stored in memory immediately into the screen memory area It needs only a small Basic program to display these screens instantly The Spectrum can produce high resolution pictures, but it takes a long time This program will not speed up that process but at least they can be called up fairly rapidly

A screen of data on the Spectrum is 6912 bytes long so starting at the top of memory and subtracting, we end up with the following addresses 58624 51712 44800, 37888 30976 The machine code is 12 bytes long giving us address 30964 So to reserve the space in memory we CLEAR 30963

I used "prog 1" to load the machine code and if all has gone well on running it the result shown should be printed The machine code is based on the LDIR instruction which will perform a transfer of a block of memory from one place to another BC is loaded with the length of the block HL with the address the block starts at and DE with the destination address So BC is loaded with 6912 HL with 58624 — the first address of our screens and DE is loaded with 16384 the starting address of the display file. A RANDOMISE USR 30964 will now call up this machine code

When this space has been reserved and the code entered it is possible to load up to five different high resolution screens into memory This is done by using a pre-recorded screen and using the direct command LOAD "" CODE 'address Where address can be one of the five mentioned previously To move a different screen to the one at location 58624 we must change the value of HL So different values must be poked directly to addresses 30968 and 30969 Fortunately the length of the Spectrum s display file is an exact multiple of 256 so we can leave address 30968 at '0 and poke 30969 with the required value These are 229, 202, 175 148 121

The driver program will (from line 2) display a different screen every few seconds depending on the *Pause* value in line 5 Once all five screens have been entered above Ramtop they can be stored on tape by SAVE "name" CODE 30964 34572 What I did was to save "SL line 1 the driver program, just before all the code so that it would load and run the code automatically

It is possible to lower Ramtop even further and get another screen in but this leaves only enough room for about three lines of Basic! Alternatively Ramtop could be raised to store the minimum required number of screens This program allows a high resolution screen to be instantly available in an ordinary Basic program and so it does not have to be loaded in separately at the beginning directly on to the screen

```
  1 REM
       SCREEN LOADER
Five screens may be loaded, at
locations 58624  51712  44800
37888  30976 using the command
LOAD ""  CODE (location).

  10 CLEAR 30963
  20 FOR n=30964 TO 30975  READ
a  POKE n,a
  30 PRINT n,  PEEK n  NEXT n
  40 DATA 1,0,27
  50 DATA 33,0,229
  60 DATA 17,0,64
  70 DATA 237,176
  80 DATA 201

30964 1
30965 0
30966 27
30967 33
30968 0
30969 229
30970 17
30971 0
30972 64
30973 237
30974 176
30975 201
30964 LD    BC,6912     1   ?
                          0   ?
                          27  ?
30967 LD    HL,58624     33  !
                          0   ?
                          229 RESTORE
30970 LD    DE,16384     17  ?
                          0   ?
                          64  @
30973 LDIR                237 GO SUB
                          176 VAL
30975 RET                 201 <1

30964 LD    BC,6912
30975 RET
30976 NOP
30977 NOP
```

```
30967 LD    HL,58624
30970 LD    DE 16384
30973 LDIR
    8>REM

    1 BORDER 0  PAPER 0  INK 7  C
LEAR 30963  PRINT "  LOADING LE
AVE TAPE RUNNING  "  LOAD ""COD
E
    2 DATA 176,000,229 200,175,14
8 121,148,1
    3 READ a  IF a=1 THEN RESTORE
  GO TO 3
    4 POKE 30969,a  RANDOMIZE USR
30964
    5 PAUSE 5  GO TO 3
```

Screen store
by Keith Robertson

Polar plotting

on BBC Micro

This program is written in Basic for a BBC Microcomputer with 32K of Ram. It uses Mode 2 to produce a series of shapes with the high-resolution graphics. The computer will draw screens of circles, ellipses, spirals and flowers.

Between each screen there is a brief pause, the screen will then clear and the next set of shapes will be drawn. Pressing *Escape* at any point will end the program, otherwise it will loop continuously.

Program notes

50 to 90 Initialise — Calls PROCintro which prints a brief introduction. ON ERROR set by line 790, the cursor is turned off by line 70 and a graphics window is defined in line 80.

100 to 580 Main loop — Draws screenful of each of the shapes, each one being separated by a delay of several seconds. PROCplot is called to do all the drawing.

600 to 770 PROCplot — This procedure controls all of the plotting used to draw the various shapes. Eight parameters are passed from the main loop to this procedure. The first is the polar equation of the shape to be plotted. The other parameters control the size of the shape, its position on the screen and whether it is to be filled in or not. Lines 640 to 690 is the loop that converts each polar co-ordinate supplied from the equation into ordinary X Y co-ordinates. Lines 710 to 760 fill in the shape if required, i.e. if FL1% is passed as true.

780 to 930 PROCintro — Initialise

940 PROCwait — Provides delay of required number of seconds

The technique used to draw all the shapes is that of polar plotting, which allows points to be represented by a distance and an angle rather than two distances. All this does is allow complex shapes to be represented by simple equations, i.e. the equation of a spiral is r = theta.

The program is quite slow, since it is written in Basic, however it does produce some nice effects. With Rem statements removed it occupies under 2K.

```
10 REM Polar Plotting Demonstration
20 REM Written for the BBC MICRO
30 REM Model B by M.J. Dunn
40 REM Initialise
50 MODE 7 PROCintro
60 MODE 2
65 REM Turn off cursor
70 VDU 23;11,0;0;0;0
75 REM Define graphics window
80 VDU 24,0;0;1279;975;
90 REM Main Loop
100 REPEAT
110   COLOUR 1 PRINT TAB(6);"CIRCLES"
120   FOR N%=1 TO 8
130     GCOL 1 RND(7)
140     PROCplot("2",RND(1279),RND(1023),RND
    (100)+40 1,2,TRUE,FALSE)
150     NEXT
160   PROCwait(3)
170   CLG
180   FOR N%=1 TO 8
190     GCOL 1 RND(7)
200     PROCplot("2" RND(1279),RND(1023),RND
    (100)+40,1,2 FALSE,TRUE)
210     NEXT
220   PROCwait(3)
230   CLS
240   COLOUR 2:PRINT TAB(6);"ELLIPSES"
250   FOR N%=1 TO 8
260     GCOL 1,RND(7)
270     PROCplot("3/(2+COS(theta))",RND(1279),
   RND(1023),RND(100)+40,1 2,TRUE,FALSE)
280     NEXT
290   PROCwait(3)
300   CLG
310   FOR N%=1 TO 8
320     GCOL 1 RND(7)
330     PROCplot("3/(2+COS(theta))" RND(1279)
   RND(1023),RND(100)+40 1 2 FALSE TRUE)
340     NEXT
350   PROCwait(3)
360   CLS
370   COLOUR 3:PRINT TAB(6);"SPIRALS"
380   FOR N%=1 TO 7
390     GCOL 0,N%
400     PROCplot("theta",640,512,10,N%,4,FALSE,
   FALSE)
410     NEXT
420   PROCwait(3)
430   CLG
440   FOR N%=1 TO 7
450     GCOL 1,N%
460     PROCplot("theta",640,512 10,N%,4,FALSE,
   TRUE)
470     NEXT
480   PROCwait(3)
490   CLS
500   COLOUR 4 PRINT TAB(5);"FLOWERS"
510   FOR N%=1 TO 8
520     GCOL 0 RND(7)
530     A%=RND(15)+4
540     PROCplot("1+COS(theta*A%)",RND(1279),
   RND(1023) 100 1,2,FALSE,FALSE)
550     NEXT
560   PROCwait(3)
570   CLS
580   UNTIL FALSE
590 END
600 REM Procedure to plot shapes
610 DEFPROCplot(eqn$ X%,Y%,SF S,N%,FL%,FL2%)
620 LOCAL theta,r,x,y,x1% y1%
630 IF FL2% THEN x1%=RND(200)-100 y1%=RND(200)-100
640 FOR theta=0 TO N%*PI STEP .063
650   r=(EVAL(eqn$)*S)
660   x=r*COS(theta)*SF+X% y=r*SIN(theta)*SF+Y%
670   IF theta=0 THEN MOVE x,y ELSE DRAW x,y
680   IF FL2% THEN PLOT 1,x1% y1% MOVE x,y
690   NEXT
700 IF NOT FL% THEN ENDPROC
705 REM Fill Shape
710 MOVE X% Y%
720 FOR theta=0 TO N%*PI STEP .063
730   r=(EVAL(eqn$)*S)
740   MOVE r*COS(theta)*SF+X%,r*SIN(theta)*SF+Y%:
  PLOT 85,r*COS(theta+.063)*SF+
  X%,r*SIN(theta+.063)*SF+Y%
750   MOVE X% Y%
760   NEXT
770 ENDPROC
780 DEFPROCintro
790 ON ERROR MODE 7 END
800 VDU 23;11 0;0;0;0
810 V=RND(-TIME)
820 PRINT TAB(6,6);CHR$(141)CHR$(131);"GRAPHICS
   DEMONSTRATION";TAB(6 7);CHR$(1
  41)CHR$(131);"GRAPHICS DEMONSTRATION"
830 PRINT
840 PRINT " This program draws on the screen a
   series of geometrical figures such as
   circles ellipses,spirals etc."
850 PRINT
860 PRINT "After each screen there will be a
   short pause,the screen will clear
    and the nextsection will be drawn."
870 PRINT
880 PRINT "Press";CHR$(128);"ESCAPE";CHR$(135);
   "to halt the program "
890 PRINT
900 PRINTTAB(6);CHR$(134);"Press any key to
   start"
910 *FX 15,1
920 A=GET
930 ENDPROC
935 REM Delay Procedure
940 DEFPROCwait(S) LOCAL T T=TIME REPEAT UNTIL
   TIME>T+S*100 ENDPROC
```

Polar plotting
by M Dunn

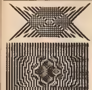
Hypnotist

on Spectrum

This compact program gives an infinite array of changing patterns.

As you can see from the examples the pattern is generated on plain and vertically striped backgrounds, thus giving differing effects. Sound is produced at the end of each drawing sequence.

To copy on to the printer, press *'Break – Copy'* and when printed *Continue*. The pattern will then commence from the last drawing sequence. Try alternative patterns by adjusting line 40.





```
10 REM      HYPNOTIST
       BY PAUL REYNOLDS
20 BORDER 0: GO SUB 120
30 LET a=170: LET c=84: OVER 1
40 LET b=INT (RND*10)
50 FOR n=1 TO 8+b: LET d=48
60 PLOT d+b,d: DRAW b,b
70 PLOT d+a-b,d: DRAW -b,b
80 PLOT d+b,d+c: DRAW b,-b
90 PLOT d+a-b,d+c: DRAW -b,-b
100 NEXT n: BEEP .1,b-40
110 GO TO 30
120 PAPER 0: CLS: INK 1+6*RND
130 IF RND<.5 THEN RETURN
140 FOR a=0 TO 87
150 PRINT ";CHR$ 133
160 NEXT a
170 RETURN
```

Hypnotist
by Paul Reynolds

Bird and catarpillar

on Vic 20

A hungry caterpillar is crawling over your screen. The caterpillar spots a nice piece of lettuce and it is up to you to see that it gets the lettuce. You have full control over the direction in which the caterpillar moves. The direction can be changed by pressing one of four keys as follows
Z for LEFT
X for RIGHT
/ for UP
for DOWN

Danger lurks. The caterpillar must not hit the wall (the edge of the screen display) otherwise it gets squashed. Also the caterpillar mustn't suddenly go backwards, otherwise it bites itself and the game ends. Thus, for example, if the caterpillar is going down don't press / for up, press Z or X first. As soon as a piece of food is eaten another piece appears.

There is more danger around. A bird is flying around the screen, it may eat the caterpillar or the food. The bird usually heads straight towards the food and hovers around the food, waiting for you. If you are fast you'll be able to make the caterpillar eat the food and escape from the bird. You'll do well if the caterpillar eats more than ten bits of food.

The program will run on any Vic20, expanded or not, lines 11 and 12 take care of the necessary changes. The many *Rem* statements explain the program.



```
1 REM          XXXXXXXXXXXXXXXXXXXXXXXX
2 REM          X                      X
3 REM          X BIRD AND CATERPILLAR X
4 REM          X                      X
5 REM          X BY  CZES  KOSNIOWSKI X
6 REM          X                      X
7 REM          XXXXXXXXXXXXXXXXXXXXXXXX
8 REM
9 REM
10 REM XXXXXXXX INITIAL SETTINGS XXXXXXXXX
11 PP=7680+(PEEK(44)>=18)*3584
12 GG=38400+(PEEK(44)>=18)*512
13 VO=36877: SO=VO-1
14 POKE VO+1 26
20 REM XXXXXXX CONTROLS XXXXXXXXXXXXXXXX
21 PRINT CHR$(147)
22 PRINT " BIRD AND CATERPILLAR  "
23 PRINT "CONTROLS " PRINT
24 PRINT "Z  LEFT          / UP"
25 PRINT "X  RIGHT           DOWN" PRINT
26 PRINT " PRESS ANY KEY TO GO"
27 GET G$ IF G$="" THEN 27
30 REM XXXXXXX SETTINGS FOR EACH NEW GAME
31 REM XXXXXXXX CATERPILLAR XXXXXXXXXXXXX
32 A(0)=PP+220
33 A(1)=A(0) 1
34 A(2)=A(1) 1
35 X=1 Y=0
37 SA=0
40 REM XXXXXXX BIRD XXXXXXXXXXXXXXXXXXX
41 B=PP+10
42 U=0 V=1
50 REM XXXXXXX SCREEN XXXXXXXXXXXXXXXXX
51 PRINT CHR$(147)
52 FOR J=0 TO 20
53 POKE GG+22*J 2
54 POKE PP+22*J 160
55 POKE GG+22*J+21,2
56 POKE PP+22*J+21 160
57 FOR I=1 TO 20
58 POKE GG+I+22*J,5
59 NEXT NEXT
60 REM XXXXXXX FOOD XXXXXXXXXXXXXXXXXXX
61 K=PP+463+INT(RND(1)*20)
62 POKE K 88
63 M=INT((K-PP)/22)
64 N=K-PP-22*M
70 REM XXXXXXX START XXXXXXXXXXXXXXXXX
71 REM XXXXXXX CATERPILLAR CONTROLS XXXXXX
72 GET A$
73 IF A$="Z" THEN X=-1 Y=0
74 IF A$="X" THEN X=1 Y=0
75 IF A$="   THEN X=0 Y=-1
76 IF A$="   THEN X=0 Y=1
80 REM XXXXXXX HAS CATERPILLAR HIT WALL? X
81 W=A(0)+X+22*Y-PP+1
82 IF W>506 OR W<3 THEN Z=1 GOTO 170
83 WA=W-22*INT(W/22)
84 IF WA=1 OR WA=0 THEN Z=1 GOTO 170
90 REM XXXXXXX HAS CATERPILLAR BIT ITSELF?
91 IF A(0)=A(2) THEN Z=2 GOTO 170
100 REM XXXXXXX THE CATERPILLAR XXXXXXXXX
101 POKE A(2) 32
102 A(2)=A(1)  A(1)=A(0)
103 A(0)=A(0)+X+22*Y
104 POKE A(0) 160
110 REM XXXXXXX HAS CATERPILLAR EATEN FOOD?
111 IF A(0)=K THEN SA=SA+1 GOSUB 200
120 REM XXXXXXX RANDOM CHANGE OF BIRD S DIRECTION
121 IF M>INT(B/22) THEN V=ABV GOTO 124
122 IF M=INT(B/22) THEN U=1 V=0
123 V=-ABV
124 IF N>B THEN U=ABU GOTO 130
125 IF N=B THEN U=0 V=1
126 U=-ABU
127 IF M>INT(B/22) THEN V=ABV GOTO 129
128 V=-ABV
129 ON INT(RND(1)*5) GOSUB 220
130 REM XXXXXXX MAKE SURE BIRD DOES NOT CRASH
131 B=B+U+22*V PP+1
132 BB=B-22*INT(B/22)
133 IF B>506 OR B<1 OR BB<2 OR BB>21 THEN U=-U V=-V
140 REM XXXXXXX THE BIRD XXXXXXXXXXXXXXXX
141 POKE B 32
142 B=B+U+22*V
143 POKE B,81
150 REM XXXXXXX HAS BIRD EATEN CATERPILLAR
151 FOR I=0 TO 2
152 IF B=A(I) THEN Z=3 GOTO 170
153 NEXT
160 REM XXXXXXX HAS BIRD EATEN FOOD XXXXX
161 IF B=K THEN GOSUB 200
162 GOTO 72
170 REM XXXXXXX CATERPILLAR DEAD XXXXXXXX
171 POKE SO 223
172 FOR I=15 TO 0 STEP -2
173 POKE VO I
174 FOR J=1 TO 100 NEXT
175 NEXT
176 POKE SO 0
177 POKE VO 0
178 PRINT CHR$(147)
179 PRINT " BIRD AND CATERPILLAR  " PRINT
180 PRINT " THE CATERPILLAR ATE    " SA
181 IF SA=1 THEN PRINT "BIT OF FOOD" GOTO 183
182 PRINT "BITS OF FOOD"
183 PRINT
184 IF Z=1 THEN PRINT " CATERPILLAR SQUASHED
185 IF Z=2 THEN PRINT "CATERPILLAR BIT ITSELF "
186 IF Z=3 THEN PRINT " BIRD ATE CATERPILLAR
187 FOR I=1 TO 1000 NEXT
188 POKE 198 0
189 PRINT PRINT " ANOTHER GO"  Y OR N   "
190 GET G$   IF G$=""  THEN 190
191 IF G$="Y" THEN 30
192 END
200 REM XXXXXXX FOOD HAS BEEN EATEN XXXXX
201 REM XXXXXXX MAKE NOISE XXXXXXXXXXXXXX
202 FOR I=15 TO 0 STEP -1
203 POKE SO,25+I
204 POKE VO 1
205 FOR J=1 TO 10 NEXT
206 NEXT
207 POKE SO 0
208 POKE VO 0
210 REM XXXXXXX NEW FOOD XXXXXXXXXXXXXXX
211 M=INT(RND(1)*23
212 N=INT(RND(1)*20)+1
213 K=M*22+N+PP
214 IF K=B THEN 211
215 FOR I=0 TO 2
216 IF A(I)=K THEN 211
217 NEXT
218 POKE K 88
219 RETURN
220 U=-U V=U RETURN

READY
```

Bird and caterpillar
by Czes Kosniowski


PIMANIA
A VERY MERRY CHRISTMAS & A HAPPY NEW YEAR


*SPECTRUM MONITOR
*SPECTRUM EDITOR/ ASSEMBLER
*ZX81 SCREEN KIT 1
*ZX81 ZX-MC *ZX81 REMLOAD
FAST MAILORDER-SEND TODAY!
£7 50
WE CANNOT FULLY DESCRIBE THIS IMPORTANT UTILITY HERE AND ASK YOU TO
SEND A S A E FOR COMPLETE DETAILS OF THIS AND ALL OUR PROGRAMS £8 50
£5.70
£7 50
£6 95
SEND S A E NOW FOR DETAILS
5 Corkscrew Hill, West Wickham,
Kent BR4 9BR
PICTURESQUE

Storing data above ramtop

***Kevin Griffiths** explains how to transfer data between programs on the 16K ZX81*

The programs in this article show how to *Load* separate data files from cassette into the ZX81 while a program is already in the machine by storing data above ramtop. All the programs require a 16K Ram pack.

On many occasions it may be beneficial to transfer data used in one program to another, so that it may be handled in a different format. Let us take an example. Suppose you were selling software and you wanted to store customers' records on computer, eg create a datafile, produce labels to stick on the packages, produce a cheque schedule for the bank and update your computer-stored accounts. If you received say, 50 orders on a given day, then you would need to type in name, address, cheque number and amount of each order into four separate programs. A far less time consuming and daunting task would be to type the information in once and pass it into each program.

Here are one program and two routines to enable you to do just that. The two routines are included within two example programs.

Program 1 will be repeated each time you use the technique. It would be advisable therefore to type in this program and *Save* it on to tape before following the example.

We are going to use a simple telephone directory, which will contain just five records as a demonstration. The directory will use the following arrays *A$*(5,10), *B$*(5,50), *C*(5) and the string *Z$*. These have been used to show that any type of data may be passed.

First enter program 1. Then add the following lines to the beginning of the program:

```
10 DIM A$(5,10)
20 DIM B$(5,50)
30 DIM C(5)
40 LET Z$ = (6 spaces) TELEPHONE DIRECTORY
   (7 spaces)
```

Any arrays which you *Dimension* must always be at the beginning of the program for this technique to work (if you wanted machine code routines you would need to store them in an array instead of a Rem line).

After entering the above lines, type Run followed by *Newline*. The number of bytes that will need to be made available to hold your data above ramtop should appear on the screen. If you are satisfied with your arrays, do as the computer asks and type *Y* followed by *Newline*; if not type *N* and correct your arrays. After typing *Y* the computer will automatically *New* the program and the K cursor will appear in the bottom left-hand corner. Now enter program 2.

The beginning of the second program already contains our *Dimensioned* arrays. This program would normally be your data entry type program. If you Run the program the computer will ask for name, address and telephone number five times. As it does so, you should invent some data and enter it. On completion, the computer will go into *Fast* mode and store a copy of your data above ramtop. Having done so, it will ask you to *New* and enter the next program. Lines 200 to 290 are the lines you would need to add to your data entry program.

The next program will recall the data and handle it as necessary, eg print labels. In our example, this program is merely going to print the data that we have entered. However, before you enter program 3, type in as a direct command

PRINT Z$

or

PRINT A$(2)

On both occasions the computer will return a report code of 2/0, proving that it cannot find the data. Now enter program 3 and simply type Run followed by *Newline*.

Again the computer will go into *Fast* mode and, after a few seconds, will return to *Slow* mode and print the data on the screen.

The routine for recovering the data is between lines 60 and 120 and must be entered in any program which needs to access the data. Note the word access, as this is all the program does. It copies the data from above ramtop, it does not destroy it so all you need to do is keep *Loading* programs with the above routine to keep using the same data.

The most important point to remember is that you must *Dimension* your arrays at the beginning of each program and in the same order. Programs 1 and 2 *Dimensioned Z$* using a *Let* statement. *Z$* was 32 characters long and contained the title. Program 3 however just defined *Z$* as an empty string 32 characters long. This is necessary for the computer to have an area to put the title in when recalling data from above ramtop.

The program and routines are simple to use and a little bit of thought by the user about program ideas and design can open up endless possibilities. ■

PROGRAM 1.

```
 200 LET A=PEEK 16400+256*PEEK 1
6401
 210 LET B=PEEK 16404+256*PEEK 1
6405
 220 LET C=B-A
 230 PRINT "YOUR DATA WILL REQUI
RE ";C-7
 240 PRINT "BYTES ABOVE RAMTOP."
 245 PRINT
 250 PRINT "IF YOU WISH RAMTOP T
O BE SET"
 260 PRINT "TO ACCOMODATE YOUR A
RRAYS THEN"
 270 PRINT "PLEASE TYPE ""Y"", T
HEN LOAD"
 280 PRINT "NEXT PROGRAM."
 290 PRINT
 300 PRINT "IF YOU WISH TO CHANG
E THE ARRAYS"
 310 PRINT "IN ANY WAY PLEASE TY
PE ""N"""
 320 PRINT "AND AMEND."
 330 INPUT Z$
 335 IF Z$<>"Y" THEN CLS
 340 IF Z$<>"Y" THEN LIST
 350 LET X=C-7
 360 POKE 16389,INT (105-(X/256)
)
 370 POKE 16388,(256*(128-PEEK 1
6389))-X
 380 NEW
```

PROGRAM 2

```
  10 DIM A$(5,10)
  20 DIM B$(5,50)
  30 DIM C(5)
  40 LET Z$="      TELEPHONE DIR
ECTORY"
  50 FOR I=1 TO 5
  55 CLS
  60 PRINT "NAME ";I
  70 INPUT A$(I)
  80 PRINT
  85 PRINT A$(I)
  87 PRINT
  90 PRINT "ADDRESS"
 100 INPUT B$(I)
 110 PRINT B$(I)
 120 PRINT
 130 PRINT "TEL. NO."
 140 INPUT C(I)
 160 PRINT C(I)
 170 NEXT I
 190 CLS
 200 REM THE SECTION FOLLOWING
         TRANSFERS THE DATA
         ABOVE RAMTOP.
 205 FAST
 210 LET A=32768-(PEEK 16388+256
*PEEK 16389)
 220 LET B=PEEK 16400+256*PEEK 1
6401
 230 FOR X=0 TO (A-1)
 240 POKE (32768-A)+X,PEEK (B+X)
 250 NEXT X
 260 SLOW
 270 PRINT "DATA HAS BEEN STORED"
 280 PRINT "PLEASE ""NEW"" AND L
OAD NEXT"
 290 PRINT "PROGRAM."
```

PROGRAM 3.

```
  10 DIM A$(5,10)
  20 DIM B$(5,50)
  30 DIM C(5)
  40 LET Z$="                    "
  50 REM    THIS SECTION RECALLS
            THE DATA TO BE USED
            IN THE REST OF THE
            PROGRAM
  60 FAST
  70 LET A=32768-(PEEK 16388+256
*PEEK 16389)
  80 LET B=PEEK 16400+256*PEEK 1
6401
  90 FOR X=0 TO (A-1)
 100 POKE B+X,PEEK (32768-A+X)
 110 NEXT X
 120 SLOW
 130 REM    EXAMPLE PROGRAM
 135 CLS
 140 FOR I=1 TO 5
 150 PRINT Z$
 160 PRINT
 170 PRINT A$(I)
 180 PRINT
 190 PRINT
 200 PRINT B$(I)
 210 PRINT
 220 PRINT "TEL. NO. ";C(I)
 230 FOR P=1 TO 50
 240 NEXT P
 250 CLS
 260 NEXT I
```

At your command

***David Nowotnik** explains how you can Peek and Poke to the Spectrum display file.*

Because of the complex layout of the Spectrum display file, the handbook suggests that you are unlikely to want to use *Peek* or *Poke* to this area of Ram. However, for high resolution interactive games or animation effects you probably will want to use these commands. The problem lies in the calculation of addresses in the display file from row and column data.

To demonstrate the order in which the display file is arranged, try this one-line program. It can be entered as a direct command

```
FOR I = 16384 @ 22527 POKE I BIN 11111111
NEXT I
```

The Bin number causes all pixels to be Ink. A mixture of 0s and 1s will produce a striped pattern. You should notice several things from this routine

1 The display file is divided into three groups of eight character rows each
2 Each character square is made up of eight rows of pixels
3 Each character square is also eight pixels across, this eight pixel row forms one byte in the display file
4 In each group of eight rows, the top pixel row of all character squares is filled in first, then the second row, and so on

To be able to calculate addresses, this pattern has to be expressed mathematically. One method of doing this is to turn to binary arithmetic. Expressing display file addresses as a 16-bit binary number I found that certain groups of bits controlled certain aspects of the screen position corresponding to that address. This is demonstrated in figure 1

Fig 1 Groups of binary bits within the screen address

```
0 1 0  0 0  0 0 0  0 0 0  0 0 0 0 0
  1     2     3      4        5
```

Group
1 — Bit 14 is set to indicate values above 16383
2 — These two bits hold values 0 1 or 2, they indicate which group of eight rows
3 — Pixel row number within a character (0-7)
4 — Character row number within a group (0-7)
5 — Column number (0-31)

From this relationship I obtained the following expression for calculating screen addresses

Address = 16384 + 32 * (y AND 192) + 256 * (y AND 7) + 4 * (y AND 56) + x

Where *y* is the pixel row number (0-191) and *x* is the byte column number (0-31). Position 0 0 is at the top left of the screen.

Unfortunately the Spectrum does not carry out conventional *And* or *Or* operations (unlike the ZX80 and ZX81), so this routine will not work using the Spectrum *And*. To overcome this problem, I turned to three simple machine code routines to perform *And* in the demonstration program in figure 2

Written for the 16K Spectrum, the *Read/Data* lines (100-140) *Poke* the machine code into the UDG area. Lines 140-180 set columns in the attribute file to random ink colours whilst lines 200-230 draw a random high-resolution bar chart. The length of the bars grows one pixel row at a time as you watch the display. The subroutine in lines 20-30 *Pokes* the *y* value into the three machine code routines, line 30 calculates the addresses according to the aforementioned formula. All the machine code does is

```
LD      B,0
LD      A,y
AND     7
LD      C,A
RET
```

similarly for *And* 56 and *And* 192 ■

fig 2. Barcharts

```
 10  GOTO 100
 20  POKE 32747,y: POKE 32755,y:
     POKE 32763,y
 30  LET a=16384+32* USR 32760+
256* USR 32744+4* USR 32752+x
 40 RETURN
100 FOR i=32744 TO 32767
110 READ a: POKE i,a
120 NEXT i
130 DATA 6,0,62,0,230,7,79,201,
6,0,62,230,56,79,201,6,0,62,0,
230,192,79,201
140 FOR i=22528 TO 22559
150 LET a=56+ RND*7
160 FOR j=0 TO 767 STEP 32
170 POKE i+j,a
180 NEXT j: NEXT i
200 FOR x=0 TO 31
210 FOR y=190 TO INT(RND*120)
STEP -1
220 GO SUB 20: POKE a,254
230 NEXT y: NEXT x
```



Plotting data according to scale

***G Morton** presents a graph plotting routine to represent data on an x, y scale*

I devised this program to enable me to quickly plot the results obtained from electrical experiments

Lines 10-100 input the experimental results in the form *x,y* Lines 110-150 and lines 160-200 determine the peak values of *x* and *y* respectively so as to be adle to scale the screen asis correctly Lines 210-220 determine the scaling factors for the *x* and *y* asis

Lines 240-270 are required because the Dragon cannot print text to the graphics screen These lines merely tell the operator the values corresponding to the graduations on the axes

Line 280 gives the operator time to read the previous screen Please note that the print statements have been laid out correctly to fill the lines without breaking any words, so don't miss the spaces Line 280 could be changed to a *press "a" to continue* form but I found the time allowed quite adequate

Lines 310-340 adjust the data by using the scaling factors Lines 330-340 change the data to integer form suitable for the *Pset* statements This produces an error of less than ½ percent quite suitable for experimental data

Line 350 prints the data to the screen while lines 370 380 plot the x and y axis Lines 390-440 plot the graduations on the asis

Line 290 defines the mode as *3* This is not the highest definition, but does allow the simultaneous plotting of several sets of data in different colours

While I do not think this is the most efficient method of setting out the program it is quite quick enough for this purpose If required, an added line at 355 could be used to plot lines between each data point.

For more than 40 points of data change the dimension statements in line 30

```
10 CLS1
20 INPUT "HOW MANY POINTS ? MAXIMUM OF 40";L
30 DIM A$(40),B$(40)
40 FOR M=1 TO L
50 CLS
60 PRINT "INPUT X COORDINATE OF POINT ",M
70 INPUT A$(M)
80 PRINT 'INPUT Y COORDINATE OF POINT ";M
90 INPUT B$(M)
100 NEXT M
110 0=VAL(A$(1))
120 FOR M=2 TO L
130 IF VAL(A$(M))>B THEN GOTO 140 ELSE GOTO 150
140 B=VAL(A$(M))
150 NEXT M
160 C=VAL(A$(1))
170 FOR M=2 TO L
190 IF VAL(B$(M))>C THEN GOTO 190 ELSE GOTO 200
190 C=VAL(B$(M))
200 NEXT M
210 D=230/B
220 E=170/C
230 CLS
240 PRINT"THE FOLLOWING GRAPH   REPRESENTS THE
    DATA PREVIOUSLY DEFINED"
250 PRINT"THE PEAK VALUE OF X IS ",0
260 PRINT"THE PEAK VALUE OF Y IS ";C
270 PRINT"      HENCE EACH LINE REPRESENTS
    1/10TH OF THESE ANSWERS ON THE  RESPEC
    TIVE SCALES"
200 FOR S=1 TO 8000 NEXTS
290 PMODE 3,1:SCREEN 1,0:PCLS
300 FOR M=1 TO L
310 A$(M)=STR$(D*(VAL(A$(M))))
320 B$(M)=STR$(E*(VAL(B$(M))))
330 X=INT(VAL(A$(M)))
340 Y=INT(VAL(B$(M)))
350 PSET(X+22,170-Y,3)
360 NEXT M
370 LINE(22,170)-(22,0),PSET
380 LINE(22,170)-(252 170),PSET
390 FOR F=1 TO10
400 LINE(230*F/10+22,172)-(230*F/10+22,168),
    PSET
410 NEXT F
420 FOR F=1 TO 10
430 LINE(20 170*F/10)-(24,170*F/10),PSET
440 NEXT F
450 GOTO450
```

In principle it's easy

This is the last article in our current series on machine code. Further machine code articles, programs and routines will follow shortly

To get a horizontal line, 10 characters long on the top line of the display we could execute the following code

```
      LD A, 80        3E 80     set value to be
                                  displayed
      LD B, 0A        06 0A     set loop count
      LD HL, (400C)   2A 0C 40  point to first
                                  character in dis-
                                  play file
      INC HL          23
LOOP  LD (HL), A      77        display
      INC HL          23        point to next char-
                                  acter
      DJNZ LOOP       10 FC     do it again
```

To do the same job anywhere else on the display all we need to do is alter the start value of *HL* by an appropriate offset. In principle it's easy to calculate the necessary offset. Let's think about the display file (see figure below).

If the *HL* is incremented after having been loaded from *D-file* so that it points at column 0, row 0, then we simply multiply the row number we want by 33 and add on the column number. That is

offset = row * 33 + column

Provided the row value never exceeds 7 we could use our 8-bit multiplier here. But there's a neater way

offset = row * (32 + 1) + column
= row * 32 + row + column

Despite the fact that this expression for the offset seems more complicated than the original, it has the advantage that the multiplication is now by a power of 2 (2^5), so all we have to do is shift row left 5 times to evaluate row * 32.

Now let us imagine that the row value is available in the *E* register, and the column value is in the *C*-register. We can calculate the offset like this

```
      LD B, 05        06 05
SHIFT SLA E           CB 23
      DJNZ SHIFT      10 FC
```

But it's not quite as easy as that! This piece of code shifts the *E* register contents left 5 times all right. That's fine if row * 32 is less than 255, but it could easily be more than that, and then the *E*-register will overflow.

So we need a 16-bit register. If we use *De*, the above code can be used as a basis for the routine, but there are some pieces to add on. First, we will have to make sure that *D* contains zero to begin with. Second, as bits shift left off the end of *E* we want them to appear in *D* and then shift along *D*. This will work

```
      LD D, 00        1600      clear D
      LD B, 05        06 05     load loop count into B
SHIFT SLA D           CB 22 }   shift left DE
      SLA E           CB23  }
      JRNC EOL        30 01     go to End of loop on
                                  no carry
      INC D           14        put the carry into the
                                  junior bit of D
EOL   DJNZ SHIFT      10 F7     test for end of loop
```

Now we want to add this into *HL* having first loaded it with the address of the first character in the display file

```
LD HL, (400C)  2A 0C 40
INC HL         23
ADD HL, DE     19
```

Unfortunately, what we now need to do is to add the row value into *HL*, and the copy in *E* has been destroyed by the shift operations. That's no real problem, because we presumably passed the row value from Basic by *Poking* it to a byte just before the beginning of the machine code routine in the usual way, and it's still available there. So all we have to do is zero *D*, load *E* from this byte and *Add Hl De* again. But this does prompt the question, "Was there a neater order in which to do things?"

Well, there was

```
      LD HL, (400C) 2A 0C 40 }  compute address of
                             }    first character
      INC HL         23      }  in display file
      LD D, 00       16 00   }
      ADD HL, DE     19      }  add row value to it
      LD B, 05       06 05

      as before                 compute 32 * row

EOL   DJNZ SHIFT     10—
      ADD HL, DE     19         add this into HL
      ADD HL, BC     09         add column value into
                                  HL
```

Now we simply execute the "draw a line" routine as before

```
      LD A, 80       3E 80  (or whatever)
      LD B, 0A       06 0A
LOOP  LD (HL), A     77
      INC HL         23
      DJNZ LOOP      10 FC
```

The hex codes are given below, tidied up.

There's no test in the routine to check that the line being drawn doesn't go over the right-hand edge of the display, and of course such a check should be included. Otherwise a pile of end-of-line returns could get clobbered. The easiest way of doing this would be to test whether the character we're about to overwrite is a newline. If so, don't.

This routine produces a horizontal line because of the *Inc Hl* instruction in the loop. Change *Hl* by some value other than 1, and we get different shapes. *Inc Hl* twice, and every other print position will display the character, for instance. Add 33 (decimal) into *Hl* in every loop and we get a vertical line. Add 34 (decimal) into *Hl* in each loop and we get a diagonal line.

You could have a library of such routines and simply call one whenever you want that kind of line.

Here is the complete code. This time we won't bother with addresses in the listing — they're not important (thanks, once again, to *relative* jumps)

```
      LD C, 00        0E 00
      LD E, 00        1E 00
      LD HL, (400C)   2A 0C 40
      INC HL          23
      LD D, 00        16 00
      ADD HL, DE      19
      LD B, 05        06 05
SHIFT SLA D           CB 22
      SLA E           CB 23
      JRNC EOL        30 01
      INC D           14
EOL   DJNZ SHIFT      10 F7
      ADD HL, DE      19
      ADD HL, BC      09
      LD B, 00        06 00
      LD A, 00        3E 00
LOOP  LD (HL), A      77
      LD DE, 00 00    11 00 00
      ADD HL, DE      19
      DJNZ LOOP       10 F9
```

The zero bytes underlined must be poked before calling the routine as follows

Start address + 1	starting column (e.g. 05 for column 5)
Start address + 3	starting row e.g. 07 for row 7)
Start address + 25	number of characters to be plotted (e.g. 0A)
Start address + 27	code of graphics character (e.g. 08 to ▚)
Start address + 30	value added to HL between plots (e.g. 01 for a horizontal line, 21 for a vertical line, 20 or 22 for diagonal lines)
Start address + 31	not normally used unless the value to be added exceeds 255, otherwise set to 00

Once you have loaded this up and seen what it does, think about incorporating it into Basic programs to generate, say, a series of squares. Use *Rnd* to find the top left-hand corner (column and row) and the length of side. Then *Poke* the relevant addresses in the machine code routine and call it via *Usr*. Do this four times for the four sides of the (open) rectangle. Don't forget to test the sizes to see if it will all fit on the screen! ■

```
                    1                   2                   3
Column →  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

Row    0  >□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□>
 ↓     1   □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□>
       2   □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□>
       3   □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□>
       4   □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□>
and so on
```

Reproduced from *Machine Code and better Basic* by Ian Stewart and Robin Jones (price £7.50) by kind permission of Shiva Publishing Ltd, 4 Church Lane, Nantwich, Cheshire CW5 5RQ

Is there anything about your computer you don't understand, and which everyone else seems to take for granted? Whatever your problem *Peek* it to Ian Beardsmore and every week he will *Poke* back as many answers as he can. The address is *Peek & Poke, PCW*, Hobhouse Court, 19 Whitcomb Street, London WC2 7NF.

COMPLICATIONS IN COLOUR

Julian Bowden of London SE9 writes

Q **I own a ZX81, and read with interest your article about its US counterpart the Timex 1000 Jeff Naylor said that he adapted his for use over here by by-passing the US modulator and attaching the unit from his own ZX81**

This reminded me that I had been given an Atari video games machine for a Christmas present, and of course I could not use it I contacted Atari who said that a modification would cost £28.75 I am wondering why it cost this much when a US computer can be adapted quite easily to run on British television

A You miss one important fact The ZX81 and Timex 1000 are both black and white video output Your Atari games machine is colour Therefore this requires a much more complex conversion from NSTC to PAL It would include some internal modification I am sure that Atari could do it as they have all the plans but it would be a very different matter for anyone else to attempt it

The other point is that a conversion done by a non-registered dealer would void your guarantee Unless you go to the opposite extreme and buy an NSTC compatible television, I would suggest that the only practicable way out of this situation is to send the machine to Atari

YOU WILL NEED SPECIAL INTERFACING

M Ridgeway of Taylor Road Southcourt, Alyesbury Bucks, writes

Q **Hopefully, at Christmas I will be getting an Atari 400 computer. I would like to ask some questions Can the Amber 2400 printer be used with the Atari 400 without special interfacing? Will the track ball for the Atari VCS, which is available in America, be able to fit the 400, and will it be released over here?**

Lastly, as yet you have not published any programs for the Atari Now that the price has been lowered to £199, I am sure that a lot more people will be buying it So, will you publish more programs for it in the future?

A The Amber 2400 will need special interfacing for use with either of the Atari machines A cable will be needed to interface the Amber to one of the joystick ports You will also need a special routine, which comes on cassette, to add the controls to the computer

The price of the printer is £80 40, the conversion cable and cassette is £18 34, and postage and packing is £2 95 All these prices are fully inclusive of VAT You will also receive complete instructions a spare inking ribbon and a spare roll of paper

A couple of people have asked about the track ball At the moment, Atari does not make one, either here or in America The one in the US is manufactured by an independent company A track ball is being considered for the new Atari computer based on the 400, but it is not due for release over here until well into 1983 None of the current games software employs the track ball so you would have to write you own routines

As for publishing Atari programs, we are more than happy to consider programs for any micro computer But so far we have had little response from Atari owners So how about sending some in?

IS THE FAULT AT THE MAINS

C Stenson of Military Road Pembroke Dock, Dyfed, writes

Q **I own a Vic20 which I have had for a week. However, after being on for half an hour, it constantly crashes or resets itself The power light also flashes on and off Is this a fault in my Vic or are mains fluctuations causing this.**

A I cannot see how mains fluctuations can cause this, unless you are having similar trouble with other domestic appliances in your house If you are, then you will have to call an electrician quickly

Far more likely is a fault in your Vic I have not met this problem before on the Vic which has a good record for reliability It would seem that somewhere along the line the power input is being overloaded or else there is a loose wire If the power light goes out then obviously you have lost power, which is the reason why the computer resets itself — it has the same effect as turning your machine off

The only thing that puzzles me slightly is why the computer waits half an hour before going off Is this time pretty constant, or is it variable? If it is constant then it might be a component at fault, such as a capacitor not discharging properly If the time varies a great deal, then it is more likely to be a loose wire

It would be as well to check the external wiring which in effect means checking the plug to see that a wire has not come loose If not, you will have to take your computer back to where you bought it and ask for it to be changed or repaired

CONTRAST CONTROL CUTS DAZZLE

Norman Peckett of Court Close, High Wycombe Buckinghamshire, writes

Q **I have had my Spectrum for two and a half weeks. Right from the start it has dazzled me Should the colours be less bright?**

Could you also explain to me how I can ask a question in a program, (eg with a Y/N answer) so that I can redirect the user to the beginning of the program, or the end, using the *Inkey$* function By the way, I received my Spectrum after cancelling my order and buying a Dragon for cash.

A It is most likely that the television is not set correctly which is the cause of your being dazzled It should not happen Try re-setting the contrast slightly

To use *Inkey$* all you need is a line like *If Inkey$ = "Y" Then Goto* and *If Not Inkey$ = "Y" Then Goto*

You are not restricted to *Goto*, but can use any of the statements that can be put after a *Then* for example *Print*, *Gosub*, *Let* and *Stop*, are all possible

LOADING AND SAVING NOT ZX81 PROBLEMS

R W Denney of Taunton, Somerset writes

Q **I would appreciate your advice on my ZX81 with QS 3K expansion After initial problems with *Loading* and *Saving*, I found that it worked if I took out the Ear jack when *Loading* However, after about three weeks my programs failed to *Load*. The first to go wrong were those near the memory limit**

I wrote to Sinclair Research and they sent me a printed sheet on this problem. The sheet advised that 'output from the cassette should be 2 to 4½ volts, peak to peak.' This output seems excessive — the output from my hi-fi is only 2½ volts. Although pleased with the ZX81, I am disappointed with the storage facilities I did think of buying a Spectrum, but not if the programs are going to become difficult to store, and can only be stored for a short while

A *Loading* and *Saving* remains the largest single problem with the ZX81 You do not actually say if the measures advised by Sinclair have been of any use You need not worry about the output of 4 volts from the Ear socket The output is AC and there are capacitors in the circuitry to cut down any overload Also remember that 4-4½ volts is the peak voltage — much of it is less than that

The 2½ volts from your hi-fi is probably DC to which different laws apply

As I have said on several occasions before, whatever problems you might or might not get with a Spectrum, there is no evidence that *Loading* and *Saving* will be among them

CLASSIFIED

Semi-display — £5 per single cc
Trade lineage — 20p per word
Private lineage — 10p per word

ACORN ATOM 10K plus 16K including 4K RTTY Eprom and software £150 ono 06614 3646

VIC, 8K Ram cartridge £27 ono Tel 021-440 2124 (evenings)

WOULD anyone help a desperate student, by producing a hard-copy of his program on a ZX Spectrum/Printer? Small reward paid Contact D King 10 Dove Crescent, Harwich

MEMORY UPGRADE, 16K-48K for the new P C B Spectrum. Ring 0293-36197

SPECTRUM SOFTWARE ZXtext Teletext simulation for 48K (60 pages) 32K (34 pages) or 16K (8 pages) Spectrum Full simulation including colour flashing 24-hour clock alarm tape microdrive storage save/verify load with auto-run freely mixed text and graphics (including user-defined graphics) etc Set up your own personal databank Cassette with full instructions and 12-months guarantee £5 Defender — can you save the Federation base from an all-out Romulan attack? Dramatic sound effects deep-space background fantastic user defined graphics. Defend the base using high powered phaser banks Cassette with full instructions and 12 months guarantee £4 20 All products available on 10-day delivery Details (SAE) Iain Stewart 17 Torry Drive Alva Scotland FK12 5NQ

WANTED PERSONAL COMPUTERS, all models bought for cash Morgan Camera Co 160 Tottenham Court Road London W1

BBC (32K) GAMES Breakout Missile Command Crazy Balloon Mastermind Simon Cyclic Defender Galactic Fighter Blitz All on one cassette £3 90 Excellent graphics and sound cheques or POs to Mr J Chaylor 32 Meerside Crescent Fishburn, Cleveland

DRAGON 32 and Vic20 owners Top quality dust covers £5.50 Cotswold Computers Church View Park Hill Hook Norton Oxon

OFFERS INVITED for surplus PERTEC 9 track tape transporters Model T7840 9-12 5 c/w Formatter Interface and PSU one unit, cabinet mounted one set of four items boxed Tel 06076-66639 day 0602-382902 evenings

DRAGON 32 software cassette price £2 95 Programs include 3D maze Hotrod a car race against the computer also Colour Artist Sketch Pad APG Computer Software 24 Mountain View Peel Isle of Man

VIC20 OWNERS At last! All-action arcade games for the unexpanded Vic at low prices Written entirely in m c for exciting colour graphics animation and sound Super games cassette — three games on one cassette Super Breakout Galaxians and Scrambler Full feature games with defined graphics hi-score free ships and bonus points Only £4 95 Fast delivery J P Shay 51 Meadowcroft Radcliffe Manchester

VIDEO GENIE MK I 16K software good condition £200 ono Aitken Annan (04612) 4561

DRAGON 32. Change sheet music to computer input easily with Mimumap £3 50 Cotswold Computers Church View Park Hill Hook Norton Oxon

JOYSTICKS FOR DRAGON 32 Essential to games enthusiast or graphic artist £7 each or £13 pair (+ 50p p&p) Microcare 1 Oakwold Road Rode Heath Stoke on Trent, Staffs Tel (09363) 5695

TI99/4A EXTENDED BASIC PROGS. Car Race Shooting Gallery Space Wars Bomb Run Defender Invader etc £8 50 including P&P Mr A Chalcraft 63 Queens Road North Weald Epping Essex

WABASH DISKETTES 5¼" SSSD £13 50 per box F&P free Add VAT to your order Make cheques payable to Marlowfield Ltd Hesketh Bank House 7-11 Yellow House Lane Southport Tel 0704 36082

16K VIC20 C2N CASSETTE DECK, one month old £150 Reply post only to 72 Highfield Rise Sheffield

VIC20 SOFTWARE The Raider Spacewarp Breakout all three games for the unexpanded Vic20 for just £3 95 Send cheques/po payable to A Carter Ewelme Ewen Cirencester Glos GL7 6BU

TRS80 Level II 16K memory VDU cassette software + manuals £200 ono Tel Sevenoaks 454707

FOUR INTELLIVISION CARTRIDGES, 1 Space Battle 1 Auto Racing 2 Tank Battles Good condition £14 each Tel 021 471 2567

48K SPECTRUM arcade games Three fantastic arcade games on one cassette Cheeky Chimp Spectral Panic and Spec Man Each game has on-screen score Hi-score and attract mode All games include colour sound and amazing graphics All this for just £4 95 Make cheques payable to Stephen Culley Long Acre Epperstone Road Lowdham Nottingham

ZX-SPECTRUM 16/48K A clever character designer plus six high class games including Pac Monster Galaxians Grandprix Catacombs £3 inclusive Send to Crown House Fort Argyll PA31 8RH

AC LINEAR CIRCUIT ANALYSIS on 48K Spectrum with ZX printer 30K program performs frequency and time domain analyses on large networks consisting of resistors capacitors inductors transformers transistors and amplifiers Tackles most topologies tabular or plot options (user set options) Menu driven Supplied on professional certified data cassette with manual Send £12 to R Wise 119 Richmond Road Montpelier Bristol BS6 5EP

Computer Swap 01-930 3266

Free readers entries to buy or sell a computer. Ring 01-930 3266 and give us the details

Spectrums for sale

SPECTRUM 48K as new with Horizon and Cambridge colour collection tapes £155. Also over £80 worth of software £35 only. Tel 01 749 1464 between 7 pm and 10 pm

ZX SPECTRUM 16K plus PRINTER very good condition £150 Call Redhill 61612 evenings

SPECTRUM 16K, five rolls of printer paper magazines + pre-program cassettes £120 ono Tel Worthing 211438

SPECTRUM 48K almost new with all leads + manual + sample cassettes £130 Tel 0262 301 659 after 6 pm

ZX81s for sale

SINCLAIR ZX81 16K with 32 cassettes worth over £200 sell for £80 ono Vincent Quah 01-882 5819 after 5 pm

SINCLAIR ZX81, 1K one month old, £40 ono Ibeclsem 01 527 5267

ZX81 16K plus games plus manuals Tel Crawley 517381

ZX80 Complete with power supply and leads plus manuals £30 Tel Grays (0375) 72150

ZX81 Memotek 64K Harris-Lackyer keyboard, £140 Will sell items separately if need be Tel 0628 30966 after 7 pm.

ZX81 COMPUTER. As new in use plus 16K RP tape recorder plus mike plus VDU (not brilliant but works) tapes and magazines £110 ono Tel 0942 212396

ZX81, with 16K instruction manual plus leads plus one tape £80 Tel 01-883 6417

ZX81 1K. Unwanted gift, boxed, £30 Tel 01-435 8732

16K ZX81 8 cassettes, two months old good condition £100 ono Tel Adam Chesham (0494) 785285

48K MEMO TECH Ram pack for ZX81 £26 Mr Sharp 01 732 7493

ZX81 16K, with tapes books £70 Telephone Nuneaton 327203

ZX81 16K, with software Tel Calne (0249) 812038 Mr L Delpoen

ZX81, £35 ono Three months old perfect working order upgrading to BBC Tel 061 223 1901 extension 2708 (Mr Jawad)

ZX81 16K, keyboard Reliably £70 ono Paul Baillie Battle 2717

16K ZX81 £100-worth of software plus mags and all leads £90 (Whitley Bay) 528686

ZX81 16K Sinclair built £30 of software including Asteroids 3D Monster Maze Sourcerers island as new £60 Tel 0601 40489

ZX81 + 16K Ram + printer two months old handy used £87 50 Tel 01-464 0945

ZX81, 16K (boxed) + manual + leads, also two books + game tape £60 D Haskins 20 West Road Old Colwyn Colwyn Bay Tel (0492) 515011

ZX81 16K keyboard case green screen 14" TV cassette printer learning lab magazines 70 cassette programs £180 ono David Sewell 01 898 6356 (day)

ZX81, 1K, hardly used seven months old all leads + manuals + a book of 30 programs many on tape in very good condition sell for £35 Tel 01-464 0682 (weekdays after 6 pm)

Commodores for sale

COMMODORE VIC20 14K high res Super Expander data cassette Bug Byte Vic Men game under guarantee £250 ono Tel Southam 3245

VIC20, Graphic Printer brand new unwanted gift, £160 ono Telephone Southampton 739734

VIC20 16K Ram pack £300 of software value £530 want only £300 Tel Amit 01-886 0510 after 6 pm

VIC20, + cassette unit with Blitz + AMOK cassette joystick + intro to Basic £180 ono Super Exp Ram £24 Vic 20 Graphic Printer complete with 1 000 sheets of paper 150 Vic20 programme reference guide £10 Telephone Chandlers Ford (04215) 67411 (1 pm-2 pm)

VIC20 plus data cassette and dust cover £140 super expander cartridge £25 Jelly Monster cartridge £12, programming reference guide £8 Dragon Maze game cassette £5 Tel 0843 203673 evenings

VIC20 plus cassette deck both boxed plus tapes and magazines £180 Contact Mark on Longleat 8164 evenings and weekends

THE COUNT, Commodore adventure cartridge perfect condition yours for £22 or other CBM adventure in perfect condition Tel Leamington Spa (0926) 832801 after 6 pm

COMMODORE VIC20 FOR SALE. Only £100 ono Tel Horley Surrey 3025

CARTRIDGES FOR VIC20 Road Race Star Battle and Jelly Monsters, only £16 each or 3 for £37 Tel 01 961 7158 evenings

SWAP the Count or Mission Impossible for Pirate s Cove. Tel 0494 76470 after 5 pm

VIC20. 1 month old with 12 months guarantee plus C2N cassette unit, plus introduction to Basic part one plus three software cassettes (boxed) £130 Tel S Bradshaw Rainford 3287

2001 PET Tool kit, sound boards manuals much software Basic overseas exchange for 48K Spectrum and printer or sell for £230 Tel 061-973 8849

VIC20 COMPUTER with C2N recorder as new Offers Tel Llanon Major 3871

VIC20 plus 16K RP both excellent condition as new only £180 Tel Barry (0446) 732282

CBM 3032 CBM 3040 dual disc drives CBM 3022 printer counting + word processing software £1 500 ono Tel 01 677 3077

PET 4032 (Feb 40) 32K + printer built in sound box tool kit picture kit super chip many programs including word processor £650 Tel Mr Clines Bourne End 26576

PET 2001 8K with internal cassette and screen with manuals software etc £225 Tel Gosforth Cumbria (09405) 332 (evenings)

COMMODORE PET 2001, 32K, complete with cassette deck all in mint condition £285 ono Tel 01 722 2166

COMMODORE VIC 20 plus AF Ram mother board and leads and 3K Ram £200 Tel Scott after 5 pm Tel 0482 40405

PET 4000 Series K manuals cassette deck Pet graphics 30 programs approximately cover £400 ono MRK assembler £48 ono tool kit £25 ono Michael Thompson Histon 022063

PET 32K new Rom large keyboard cassette tool kit, microassembler Basic 4 Eprom programmer software and many books £450 ono Tel Davis on 01 543 1980 (after 5 pm and weekends)

Ataris for sale

ATARI 400 new Basic + cassettes joysticks various extras £300 ono P Hawkins Bracknell (0344) 67659

ATARI VCS games system with 8 cartridges Defender Missile Command Asteroids Champion Trip Soccer Outlaw Combat. New £330 offer £140 Tel 01 967 2017

ATARI VCS Very good condition, complete with 4 cartridges Space Invaders Nightdriver Outlaw Combat, £89 Tel Reigate Surrey 44129 evenings

ATARI VCS 9 cartridges Defender Missile Command Asteroids Pacman Nightdriver Slaying Boxing Video Pinball Combat. Excellent condition £120 ono Tel 021 356 9795

ATARI VCS 7 cartridges Combat Breakout, Space Invaders, Asteroids Video Pinball Missile Command Defender £140 Tel 521196

ATARI VCS. Very good condition with 2 cartridges, Asteroids and Combat, £55 Tel 0202 863583 Mark anytime

ATARI 800 plus 48K plus Basic cartridge plus 8 cassettes (Eastern Front, Caverns of Mars, Airstrike 4T Tutelage Galactic Chase) £350 ono worth £460 Tel 01 274 7887

ATARI VIDEO GAME, Space Invaders cartridge £70 ono Tel 01-452 7048 (evenings only) Ask for Rajsan Patel

ATARI 400 16K Basic cartridge program recorder chess cartridge are manuals £250 Tel 01-883 3420

Acorns for sale

ACORN ATOM 12+12K with disk pack FP Rom toolkit and BBC Rom all books and leads, £400 Tel Falmouth 314598 (day) 312718 evenings or weekends

ACORN ATOM, 64K Ram plus 20K Rom books + software £280 ono Tel 061 747 9622

ACORN ATOM 12K Rom 12K Ram and power pack plus basic book £120 ono Tel 01-501 1629 (eves)

ACORN ATOM 12K Ram 12K Rom plus 4K utility Rom cassettes (book 3 amp power supply and £80 of software Price £130 ono Tel 061-877 7326 evenings

ACORN ATOM 8K + 12K PSU £80 of games books etc, 12 months old £100 Tel Swanage (0929) 424423 evenings

ACORN ATOM, 12K + 12K, floating point Rom leads power supply manual + approximately £90 of software including Pacman Invaders Galaxians £130 Tel Harlow 418 609

ACORN ATOM, 12K + 16K, floating point Rom plus toolbox PSU plus expansion board giving 5K, with all leads and manual condition as new £200 ono Tel 0843 582652 after 6 pm

ACORN ATOM 12K RAM + 16K ROM 3 amp p s u tape recorder manuals and software £175 00 Tel 0480-890 403

Tandys for sale

VIDEO GENIE 1 with sound and arrow key modification includes Tandy monitor and 72 programs very good condition £380 Worksop 476413.

TRS80 16K Level II Tandy cassette + Tandy monitor Teletype 33 + custom interface + some software £350 Tel 01-677 3077

VIDEO GENIE, 16K, Level II Basic + 9" green screen manual mem soft editor assembler + large selection of books manuals cassettes and software £275 ono Tel 021 745 5884

TRS80 Model 1 Level II 16K with monitor cassette recorder joystick + sound also manuals magazines and approximately 100 arcade machine language games £450 Tel Worksop 477053

GENIE 1 16K, 10 months old plus Green Screen Monitor 12in EG 101 manuals etc £350 ono Tel Stevenage (0438) 721218 after 5 pm

TANDY TRS80 Model I Level II (16K memory) expansion interface unit (additional 16K memory) disc drive (5¼" mini disc) disc drive interface books Seikosha GP100A printer printer interface cable bulk eraser disc drive head cleaner £1,280 ono (all as new) Tel (0332) 72035

TANDY SYSTEM 8K Ram Basic, lots of software Cost £300 offers around £85 Tel 0229 54768 after 6 pm

VIDEO GENIE, 16K, excellent condition £140 Tel 01-657 3000 evenings (Croydon area)

TRS80, Model 1 Level II 2½ years old 48K expansion interface high res character generator upper lower case green screen cassette rewinder £350 of software complete £800 ono Tel Long Crendon 208232

TANDY TRS80 Level II 16K, screen + cassette recorder editor assembler magazines + books £250 ono Reece Worcester 352623

VIDEO GENIE LG 3003 16K, + extra keys + sound box built in cassette deck software + magazines sell for £200 ono Telephone 01-946 1428

TRS80 Model 1 level II 16K with Green Monitor VDU 16 months old with £100 of s-ware £70 of books £250 ono Tel 08076 65930

TRS80 level I 4K, with game tape and manual 10 months old, little used £150 ono Tel Wetherby W Yorks 63564 (evenings)

GENIE 1, + £50 of s/ware 2 books, mint condition £250 Tel South Bensley 56348

TRS80, 16K level II Green monitor cassette, many programs + Tandy speech synthesiser £350 Tel Farnborough 81133

TRS80 model 1 16K, with cassette player + monitor + s-ware manuals £260 ono Tel Gary Harlow 412242

TANDY TRS 80, level 1 4K with 16 cassettes £100 Telephone 01 729 5302 (late evenings)

For sale

SEIKOSHA, GP80A PRINTER as new boxed scarcely used £130 negotiable Tel 01 552 1467 evenings

16K CASED NASCOM Neasys plus Nasbug colour teletext graphics. New pen Zeap Nasdis Debug Roms Sargon plus chess, graphics Xtal basic, extended colour command New baby as bwel ofter Tel 061 327 4684

CENTRONICS 737 PRINTER As new, Centronics cable manual £250 ono Tel Robert 01-674 5534

DRAGON 32 + £80 extras 1 month old Quick sale £230 ono Mr A Mills, Flat C 11 Agar Grove London NW1

DRAGON 32, + computer voice cassette + books 4 weeks old £175 00 Tel 01-661 1866

ACULAB FLOPPY TAPE SYSTEM Plus a number of programs suitable for early Video Genie or TRS80 model I £100 Tel 0634 66539

VOICE CHESS CHALLENGER 10 playing levels speaks its moves, 50 word vocabulary 64 pre-programmed games Excellent condition Cost over £200 new Only £70 ono Tel 01-979 4432 after 6 pm and weekends

8K GRAPHIC ROM plus 2K VDG complete with all available software £90 ZX81 16K Ditronics keyboard and £37 worth of software £900 together £110 Palmer Norwich 611133 ext 5006

DRAGON 32, 2 months old, perfect condition with case leads and two Dragon data case progs £180 Tel 0273 688658 (after 6 pm)

DRAGON 32 as new 4 months old inc cassette lead and over £30 of tapes + books £150 Tel Reading (0734) 690935 (after 6 30 pm)

TI99/4A, prog language cartridge invaders cartridge 4½ months ale worth £345 yours for £280 ono A Collier Northwood (65) 22650

SHARP MZ80K 48K plenty of programs including Space Invaders Stark Trek, Doctor Livingstone etc. Also Zen assembler plus 2 extra manuals, £360 ono Tel 01-884 5450

SHARP MZ80K 48K, 4MHZ hi-res graphics green screen software includes basic fortran, forth assembler and many games and books £425 ono Telephone 01 316 1692

BLACK AND WHITE MONITOR Motor roller based screen home-made casing suitable for BBC £50 Tel Lee Valley 710754

DRAGON 32, new £180 Telephone 01-732 4777 Mr John Aldred

FIVE ROLLS SINCLAIR PRINTER PAPER, £8 inc p&p 01 866 1444 after 7 pm

DAI COLOUR COMPUTER, 48K 24K Basic RS232 paddles cassette sound £400 or swap for Atari 800 Bognor Regis (0243) 861086

SONY C7 BETAMAX, recorder six months old swap for any computer (not Sinclair) to the value of £430 (second-hand) (Open to any offers — can offer cash) Tel 01-672 9376

SHARP MZ 80K 48K as new with guarantee £325 Tel 051 536 3078

TANGERINE COMPUTER 16K Ram two high res boards all options Apple PSU keypad + keypad extended Basic fully cased with fan lots of software Offers Tel 0483 31949

DRAGON 32, six weeks old still under guarantee £185 ono Space Invader cartridge £12. Mrs Mewlevaux 01 304 3659

TI99/4A THERMAL PRINTER for sale offers over £150 H Jones, Kerry Wales (066686) 502 (evenings and weekends)

SHARP PC1500 colour printer plotter + 8K Ram one month old hardly used, £279 Tel 01-464 0845

9" MONITOR black and white £20 ono Tel 01 301 4783

8K PACK BASIC III, cassette deck dust cover manuals games books £400 ono Tel 01-363 3801 (Nick)

VIC20 CARTRIDGE for exchange Pirate Cove, Super Lander Super Shot. Tel 0438 811634 after 6 pm

Wanted

SWAP VIC20 Adventure Golden Baton, for Time-Machine or Arrow of Death or other Vic adventure Phil McDonald, Bournemouth (0202) 882974

WANTED ZX Spectrum 16K or 48K John Ireland Cardionoe (0482) 77406

VIC20 SOFTWARE to swap Star Bat tle cartridge plus Lair 16K Adventure for any CBM Adventure cartridge except Private Cove Harland Aycliffe (0325) 316966

SWAP Adventure land cartridge for any other adventure Tel 0733-237101

ZX SPECTRUM. Any K Mr Holt. Tel 061 794 5172 evenings

ZX81 Sinclair built 16K power pack leads etc full size moving key keyboard + software £70 Tel 01 301 1492 Mr Sims evenings

SWAP VIC20 Audio genie Tank Attack cartridge for any other Commodore cartridge Tel 0772 744439 after 6 pm

ALIEN FOR NIGHTRACER or Super slot. Phone 061-497 7640 Phone between 5 15 and 6 00 pm.

SWAP VIC20 Pirate Cove cartridge for another adventure Tel Stevenage (0438) 811634 (after 6 pm)

VIC20 OR SINCLAIR (possibly extras) locally Tel Bromsgrove 75226 Mr Swan

VIC20 + cassette deck Will offer cash Tel 01 575 3779 Martin (evenings)

WANTED ZX81 16K about £50 Tel Arundel (0903) 883063 (evenings or weekends)

DRAGON OR SIMILAR, exchange musical instruments Tel Stephen after 7 pm Bradford 603500

1 QUICKSILVA CHARACTER BOARD any price considered. Tel Gregory 0348 882026

WANTED 48K SPECTRUM Mr Toerad. 01-834 7743827 (daytime)

WANTED VIC20, super expander or 3K Ram or any other cartridges in exchange for software Tel 01-888 0610 after 4 pm

WANTED TWO BOOKS from Melbourne House Publishers *The Complete Sinclair ZX81 Basic Course Machine Language Programming Made Simple* Tel 0271 84194

Ziggurat



Beautiful programs

There is a German proverb which roughly translated says Could everything be done twice everything would be done better What this boils down to is that it is easy to be wise after the event — hindsight is twenty twenty!

In computing I wonder if those critics who pontificate about this and that have ever produced an original work themselves Aristotle wrote (in *Politics*) They who are to be judges must also be performers

This introduction is intended to set the stage for some critical comments of my own about a program published in *Open Forum* I will not going to say which issue — it was some time ago — or use the exact lines from the program but I assure you that the program is no mirage It is easy to knock with the experience of hindsight, but as I have published programs myself for others to criticise (and they certainly have) perhaps I might be allowed a few observations

The program was written by a ten year old child which I think is very important To have written a program of the complexity of that child s attempt at the age of ten is commendable However at that age it is very easy to get carried away with a program and it is difficult to hold oneself back

When I was looking through the listing of the program, my attention was attracted to lines such as

```
1000 IF T = A OR U = 1 THEN PROCBANG
1010 IF T = B OR U = 2 THEN PROCBANG
1020 IF T = C OR U = 3 THEN PROCBANG
1030 IF T = D OR U = 1 THEN PROCBANG
1040 IF T = E OR U = 2 THEN PROCBANG
```

There seemed to be a rather obvious regularity The repetition consisted in the five *If* statements which all referred to the same procedure

This repetition was compounded in my search to discover the nature of the variables *A* to *E* The original assignments to the variables were contained in one line

```
10 A = RND(9) B = RND(9) C = RND(9) D =
   RND(9) E = RND(9)
```

The form was there for all to see All the five variables *A* to *E* were the same though of different value

When faced with such a display of repetition it is difficult to understand why it was not exploited in some way when the program was written The reason of course is that form and symmetry are in the eye of the beholder — and such an eye is sharpened by hindsight. The ten-year-old in question obviously did not see the program as a whole just as a collection of parts

So what is wrong with that? Nothing, except it is a very inefficient way of programming

The aforementioned program could be improved by finding a way of coping with variables which are the same yet can take different values

```
 5 DIM VAR(5)
10 FOR I = 1 TO 5 VAR(I) = RND(9) NEXT I
```

When we come to the *If* statements, we can see that the numbers to which *U* is compared have a logical pattern So we can write

```
1000 FOR I = 1 TO 5
1010 IF T = VAR(I) OR U = I - INT((I - 1)/3) * 3
     THEN PROCBANG
1020 NEXT I
```

Sad to say however as far as the routine programming one tends to see published is concerned, both the use of arrays (dimensioned variables) and modular (or clock) arithmetic is rare

The potential saving in programming space is considerable though The amount to which we can economise depends on the way we approach programming Beauty is all! ■

Boris Allan

Puzzle

A's down

Puzzle No 36

		1	2
3	4		
5			
■	■		■

Across 1 *A* — *B* 3 *A* + *B* 5 *A*²
Down 1 *B*² 2 5*B* 3 *B*; 4 *A* — *B*.

Solution to Puzzle No 32

This algorithm produces Pascal s triangle The number of families in each cave is given by the sum of the numbers of families in the two adjacent caves immediately above

```
          1            Row 0
         1 1           Row 1
        1 2 1          Row 2
       1 3 3 1         Row 3
      1 4 6 4 1        Row 4
    1 5 10 10 5 1      Row 5
```

The numbers in the rows correspond to terms in the Binomial expansion $(a + x)^n$ For example to find the terms in the fifth row we expand $(a + x)^5 = 1a^5 + 5a^4x + 10a^3x^2 + 10a^2x^3 + 5ax^4 + 1x^5$

The numbers in front of the terms (called the coefficients) give the numbers of families at each level of the cave system

The sum of the coefficients in each row gives the probability of successive tossing of a coin producing a head (or a tail) repeatedly For example — what is the probability of getting four heads in four tossings? Look at the fourth row of the triangle 1 + 4 + 6 + 4 + 1 = 16 — so the probability is 1 in 16

Winner of Puzzle No 32

The winner is W R Masefield Slade Road, Holland-on-Sea Essex who receives £10

Top 10

Atari
1(1) Preppie (Adventure International)
2(7) Scott Adams Adventures (Adventure International)
3(4) Air Strike (English Software)
4(2) Jumbo Jet Pilot (Thorn EMI)*
5(3) Submarine Commander (Thorn EMI)*
6(5) Snooker and Billiards (Thorn EMI)
7(–) Hell Cat Ace (Microprose Software)†
8( ) War (Adventure International)†
9( ) Soccer (Thorn EMI)
10( ) Snooper Troops 1 (Spinnaker)†
*Cartridge †Disc
(Figures compiled by Calisto Computers Birmingham 021 632 6458)

Spectrum
1(1) Time Gate (Quicksilva)
2( ) Spectres (Bug-Byte)
3(3) Escape (New Generation)
4(2) Orbiter (Silversoft)
5(8) Adventure 1 (Abersoft)
6(6) Football Manager (Addictive Games)*
7(10) Master File (Campbell Systems)
8(7) Espionage Island (Artic)
9(–) Night Flite (Hewson)
10( ) Golfman (Campbell Systems)
*Requires 48K
(Figures compiled by Buffer Micro Shop London 01 769 2887)

Books
1(1) Spectrum Machine Language for the Absolute Beginner *Tand* (Melbourne House)
2(5) ZX Spectrum Explored *Hartnell* (Sinclair Browne)
3( ) Assembly Language Programming for the BBC Micro *Birnbaum* (Macmillan)
4( ) BBC Micro Revealed *Ruston* (Interface)
5( ) The Spectrum Handbook, *Langdell* (Century)
6( ) Programming the 6502, *Zaks* (Sybex)
7( ) 35 Programs for the Dragon 32, *Langdell* (Century)
8( ) Over the Spectrum *various authors* (Melbourne House)
9(5) Instant Cobol and Better Basic, *Stewart and Jones* (Shiva)
10( ) Programming the Z80 *Zaks* (Sybex)
(Figures compiled by Watford Technical Books Watford 0923 23384)
(Last week s position in brackets)

ZX81
1(3) Frogger (DJL Software)
2(2) 3D Defender (JK Greye)
3(1) Mazogs (Bug-Byte)
4(4) Mazeman (Abersoft)
5(5) Gulp II (Campbell Systems)
6(8) Gauntlet (Cascumatic)
7(7) Flight Simulation (Psion)
8(8) Adventure 1 (Abersoft)
9( ) 3D Monster Maze (JK Greye)
10(9) Chess (Artic)
*All 16K.
(Figures supplied by Buffer Micro Shop London 01 769 2887)

Vic20
1(3) Traxx (Llamasoft)†
2(5) Sargon II Chess (Commodore)
3(7) Jellymonsters (Commodore)
4(2) Defenda (Llamasoft)†
5(1) Grid Runner (Llamasoft)
6(4) Abductor (Llamasoft)
7(6) Myaad (Rabbit)
8(9) Blitz (Commodore)
9(6) Adventureland (Commodore)
10–(16) Spiders of Mars (Audiogenic)
–( ) Voodoo Castle (Commodore)
*Cartridge †Requires 8K or 16K
(Figures compiled by the Vic Centre London 01-368 9904)

LOSERS

pulling the plug out,
denying the computer
a goal scoring
opportunity. OFF!

